Degree Project in Machine Learning

Second cycle, 30 credits

Evaluating Rectified Flow Against Diffusion and GANs Models for Synthetic Generation of Fixational Eye Movements

Benchmarking Rectified Flow, diffusion, and GANs for fixation eye movement generation

RAHNAMA R. SAFDARI

Evaluating Rectified Flow Against Diffusion and GANs Models for Synthetic Generation of Fixational Eye Movements

Benchmarking Rectified Flow, diffusion, and GANs for fixation eye movement generation

RAHNAMA R. SAFDARI

Degree Programme in Electrical Engineering

Date: March 2, 2026

Supervisors: Vasileios Saketos, Yiting Wang, Mattias Nilsson

Examiner: Amir H. Payberah

School of Electrical Engineering and Computer Science

Host organization: Department of Clinical Neuroscience, Karolinska Institutet.

Swedish title: Utvärdering av Rectified Flow jämfört med diffusion- och GAN-modeller för syntetisk generering av ögonfixationer

Swedish subtitle: Benchmarking av Rectified Flow, diffusion-modeller och GANs för generering av ögonfixationer

Abstract

Fixational eye movements are involuntary eye motions occurring during visual fixation and consist primarily of drift, tremor, and microsaccades. These movements have been studied as potential markers for neurological disorders such as Parkinson's disease. Despite this relevance, large scale datasets for fixational eye movement are scarce, particularly in clinical settings. This thesis addresses the problem of synthesizing realistic fixational eye movements using deep generative models. Generating physiologically plausible fixation signals is challenging because fixational eye movement exhibits both stochastic variability and structured microsaccade characteristic such as event rates and a main sequence relationship. Prior approaches often prioritize global signal similarity and downstream task, but may be limited to reproduce real microsaccade characteristics, reducing their reliability for downstream analysis and data augmentation. To investigate this problem, three generative methods were implemented, Generative Adversarial Networks, diffusion based models, and novel application of Rectified Flow. All models were trained on preprocessed positional recordings using two normalization methods, Sinusoidal normalization and Z-Score normalization, and generated samples were reverse normalized for microsaccade evaluation. Model performance was assessed through a combination of signal level and physiological metrics, including distributional similarity measures, power spectral density analysis, and microsaccade based evaluation. The results show that Rectified Flow generation generation is closest to the real data and provides the most consistent performance across evaluation metrics and demonstrates robustness to the choice of normalization method. Diffusion based generation achieves competitive performance under Sinusoidal normalization but degrades under Z-Score normalization, suggesting sensitivity to preprocessing that impacts temporal dynamics. In contrast, Generative Adversarial Networks produce noisier sequences and exhibit weaker physiological microsaccade characteristics. Overall, the findings highlight that microsaccade characteristics provide an important validity criterion beyond global similarity measures and support Rectified Flow as a promising approach for generating physiologically plausible fixational eye movement. Although Rectified Flow achieved the strongest overall performance, the generated sequences still did not fully match all microsaccade characteristics observed in real data, highlighting remaining limitations in physiological realism. Future work should therefore focus on improving microsaccade realism and validating model generalization across

datasets.

Keywords

Deep Generative Models, Rectified Flow, Flow Matching, Diffusion Models, Generative Adversarial Networks, Fixational Eye Movements, Synthetic Data Generation, Continuous-Time Generative Models

Sammanfattning

Fixeringsögonrörelser är ofrivilliga ögonrörelser som uppstår under visuell fixering och består huvudsakligen av drift, tremor och mikrosaccad. Dessa rörelser har studerats som potentiella markörer för neurologiska sjukdomar såsom Parkinsons sjukdom. Trots denna relevans är storskaliga datamängder för fixeringsögonrörelser begränsad i storlek, särskilt i kliniska miljöer. Denna arbete behandlar problemet med att syntetisera realistiska fixeringsögonrörelser med hjälp av djupa generativa modeller. Att generera fysiologiskt rimliga fixationssignaler är utmanande eftersom fixeringsögonrörelser uppvisar både stokastisk variabilitet och strukturerad mikrosaccad karakteristik såsom händelsefrekvenser och ett huvudsekvens förhållande. Tidigare metoder prioriterar ofta global signallikhet och nedströmsuppgift, men kan vara begränsade till att reproducera verkliga mikrosaccad karakteristik, vilket minskar deras tillförlitlighet för nedströmsanalys och dataförstärkning. För att undersöka detta problem implementerades tre generativa metoder, Generativa Adversarial Nätverk, diffusionsbaserade modeller och en ny tillämpning av Rectified Flow. Alla modeller tränades på förbearbetade positionsregistreringar med hjälp av två normaliseringsmetoder, sinusformad normalisering och Z-Score-normalisering, och genererade data omvänt normaliserades för mikrosaccad utvärdering. Modellprestanda utvärderades genom en kombination av signalnivå och fysiologiska mätvärden, inklusive distributionslikhetsmått, effektspektraldensitetsanalys och mikrosaccad baserad utvärdering. Resultaten visar att Rectified Flow-generering är närmast verkliga data och ger den mest konsekventa prestandan över utvärderingsmått samt demonstrerar robusthet i valet av normaliseringsmetod. Diffusionsbaserad generering uppnår nära prestanda under sinusformad normalisering men försämras under Z-Score-normalisering, vilket tyder på känslighet för förbehandling som påverkar temporal dynamik. Däremot producerar Generativa Adversarial Nätverk mer brusiga sekvenser och uppvisar svagare fysiologiska mikrosaccad egenskaper. Sammantaget belyser resultaten att mikrosaccad egenskaper ger ett viktigt validitetskriterium utöver globala likhetsmått och stöder Rectified Flow som en lovande metod för att generera fysiologiskt verkliga fixeringsögonrörelser. Även om Rectified Flow uppnådde den starkaste totala prestandan, matchade de genererade sekvenserna fortfarande inte helt alla mikrosaccad egenskaper som observerats i verkliga data, vilket belyser återstående begränsningar i fysiologisk realism. Framtida arbete bör därför fokusera på att förbättra mikrosaccad realismen och validera modellgeneralisering över datamängder.

Nyckelord

Djupa generativa modeller, Rektifierat flöde, Flödesmatchning, Diffusionsmodeller, Generativa adversariella nätverk, Fixationella ögonrörelser, Syntetisk datagenerering, Kontinuerliga tidsbaserade generativa modeller

Acknowledgments

I would like to express my thanks to everyone who supported me throughout this project, and to special thanks to my partner Emilia.

At KTH, I would like to thank my supervisor Vasileios Saketos and my examiner Amir H. Payberah for their guidance and feedback which was valuable in helping me structure the thesis and improve the clarity and presentation of the report.

At Karolinska Institutet, I am especially grateful to my supervisor Yiting Wang, whose support was essential to this work. Through our regular meetings, she helped me understand key topics, provided hands-on guidance with implementation and coding, and contributed with many helpful ideas through continuous discussions. Her support and engagement played a major role in the progress of this project. I would also like to thank Mattias Nilsson for carefully reviewing my report and providing detailed and constructive feedback. His suggestions helped strengthen both the overall structure and the theoretical content of this thesis.

The report text were revised with the assistance of an artificial intelligence language model. The use was to improve scientific tone, clarity, and conciseness through rephrasing and grammar refinement of author-written sentences. The technical content, experimental design, implementation, results and conclusions were produced by the authors.

Stockholm, March 2026
Rahnama R. Safdari

Contents

List of Figures

List of Tables

List of acronyms and abbreviations

AI	Artificial Intelligence
DCGANs	Deep Convolutional Generative Adversarial Networks
DDIM	Denoising Diffusion Implicit Models
DDPM	Denoising Diffusion Probabilistic Models
FEM	Fixational Eye Movement
GANs	Generative Adversarial Networks
JSD	Jensen Shannon Divergence
MLP	Multilayer Perceptron
ODE	Ordinary Differential Equation
PSD	Power Spectral Density
ReLU	Rectified Linear Unit
SDE	Stochastic Differential Equation
SiLU	Sigmoid Linear Unit

Chapter 1

Introduction

Over the past decade, AI has become increasingly integrated into healthcare, demonstrating remarkable capabilities across various clinical domains such as radiology, where it is able to detect tumors from MRI scans [1]. In line with this broader trend, this thesis focuses on the application of AI within the field of eye tracking for healthcare purposes. Specifically, it explores the use of deep generative models to synthesize Fixational Eye Movements (FEM).

There is evidence that eye-movement behavior reflects underlying cognitive and neurological processes [2]. In particular, FEM, including microsaccades, have been proposed as candidate biomarkers for neurological disorders such as Parkinson's disease, though their diagnostic sensitivity and specificity remain under active investigation. Unlike general eye movement during for instance reading, FEM occur continuously during visual fixation and are largely independent of external stimuli, making them well suited for studying neurological dynamics. Consequently, modeling and synthesizing FEM is of particular interest for clinical analysis and data augmentation, as they potentially capture subtle physiological indicators that may be obscured in general eye movements.

However, the practical development of these models is often constrained by the limited availability of clinical data, which can hinder model robustness and generalizability. Collecting data in clinical contexts is expensive and time-consuming and can in some cases be hindered due to privacy laws, therefore to address this issue, generative models offer a promising solution by producing realistic and diverse synthetic data, thereby addressing the problem of data scarcity.

Recent advances in state-of-the-art generative modeling have demonstrated impressive results in fields such as image and audio synthesis [3]. Building upon these developments, this work we investigates the use of three families of generative models, Diffusion Models, Generative Adversarial Networks (GANs), and Flow Models for synthetic FEM generation. While Diffusion and GANs based approaches have already been explored for scanpath synthesis, to the best of our knowledge, none have been explicitly applied to FEM focused synthesis and Flow Matching based models are novel in context. Given Flow Matching's strong performance in other domains [4], this project aims to provide a comparative evaluation of these three model types, highlighting the novel use of Flow models for FEM synthesis.

This project was conducted in collaboration with the Karolinska Institute. The primary objective was to explore the use of generative modeling for the synthesis of FEM, with the long-term goal of augmenting the limited clinical datasets available for patients with Parkinson's disease. The synthesized data are intended to support downstream research tasks such as early Parkinson's disease detection. It is important to note, however, that the focus of this thesis lies on the development and evaluation of generative models for data synthesis, rather than on implementing or training disease detection models directly.

1.1 Background

Eye tracking involves recording the eye movements of a subject while they perform a specific task. These recordings capture various gaze events such as fixations, saccades, blinks, and smooth pursuits. In this work, FEM are of particular interest because they are largely involuntary, can be measured objectively, and provide a quantitative window into the function of the oculomotor system and its underlying neural control mechanisms. However, datasets containing such eye movement sequences are often limited in size and availability. Deep generative methods address this challenge by learning to synthesize realistic samples for small datasets. Examples of state of the art generative models are; diffusion, GANs and Flow Matching which is a novel method applied to eye tracking.

Diffusion models operate by gradually corrupting data with noise over a series of time steps until it becomes nearly random. A neural network is then trained to reverse this process by predicting and removing noise step by step in order to reconstruct realistic samples. The model can subsequently generate new data by starting from pure noise and iteratively denoising it. GANs, on the

other hand, are trained in an adversarial framework consisting of two models, a generator that produces synthetic samples and a discriminator that attempts to distinguish between real and generated data. Through this competition, the generator progressively learns to produce increasingly realistic outputs.

Flow Matching models and specifically Rectified Flow are novel applications in the eye tracking domain, they learn a velocity field that defines how samples from a simple prior distribution (such as a Gaussian) are continuously transformed into samples from the target data distribution. The model is trained to approximate the true transport dynamics between these distributions, allowing for stable and efficient generation through numerical integration of the learned flow field. Rectified Flow, a refined version of Flow Matching, simplifies training by aligning the forward and reverse transport processes, improving convergence and sample quality. Recent work [5] has demonstrated its effectiveness in modeling continuous and temporal data, motivating its exploration in this thesis.

Despite recent progress in modeling scanpaths, the generation of FEM without conditioning on stimulus content, task context, or subject-specific information remains largely unexplored. This gap motivates the present work and raises the question of whether deep generative models can be used to augment limited clinical datasets through the synthesis of realistic FEM.

1.1.1 Original problem and definition

In this work our goal is to address the limited availability of FEM data in clinical and research contexts. This scarcity poses a significant barrier to the development of robust machine learning models for diagnostic tasks, such as early detection of neurological disorders. While deep generative models have previously been applied to scanpath synthesis, most notably in SP-EyeGAN [6] and DiffEyeSyn [7], these approaches focus on stimulus driven eye movements and do not directly address the synthesis of pure, unconditional FEM.

The central research question of this thesis is: **can deep generative models successfully synthesize FEM that mimic the temporal and physiological characteristics of real data, despite the random nature of FEM trajectories.** Additionally, we investigate whether different generative families vary in their ability to model FEM. To this end, a GANs architecture adapted from SP-EyeGAN, a diffusion based model adapted from DiffEyeSyn, and a Rectified Flow model are implemented and evaluated for unconditional FEM synthesis.

1.1.2 Scientific and engineering issues

From a scientific perspective, this thesis addresses two questions. First, it examines the feasibility of synthesizing FEM with real characteristics, including realistic velocity distributions, spectral profiles, and microsaccade behavior. Establishing this feasibility is necessary before synthetic FEM data can be considered for downstream use in analysis or data augmentation.

Second, given that FEM synthesis is feasible, the thesis investigates whether Rectified Flow models synthesize FEM that more closely resemble real data compared to diffusion models and GANs, as evaluated through distributional similarity, spectral analysis, and extracted microsaccade characteristics.

From an engineering perspective, the primary challenges includes:

- Adapting scanpath oriented generative models to operate on fixation only signals.
- Designing a stable training and evaluation pipeline for modeling subtle, low-amplitude time series on limited datasets.
- Implementing and validating Rectified Flow architectures for FEM synthesis, to which they have not previously been applied.

1.2 Purpose

The purpose of this work is to investigate the effectiveness of state-of-the-art deep generative models in synthesizing realistic FEM, addressing data scarcity in clinical and eye tracking research. By generating high-quality synthetic data, this project aims to enable downstream applications such as disease detection studies, which are often limited by the availability of sufficiently large and diverse clinical datasets. Synthetic data generation also helps mitigate privacy concerns associated with patient information, as it reduces the need for direct access to sensitive clinical records.

This thesis's purpose is also to contribute scientifically by evaluating and comparing different generative paradigms, particularly through the implementation of the novel Rectified Flow approach within the eye-tracking domain. The results may provide valuable insights for both clinical researchers, who can benefit from enriched datasets for neurological and cognitive studies, and Artificial Intelligence (AI) researchers, who can gain evidence on the applicability of generative modeling techniques to similar

tasks within the eye tracking domain.

In addition, this work examines the impact of different data preprocessing strategies on generative model performance. Two preprocessing pipelines are considered: Sinusoidal Normalization and Z–score Normalization. Evaluating models under both settings allows for a broader understanding of how preprocessing choices influence FEM synthesis.

Finally, this project also serves as part of the completion requirements for the Master's program in Machine Learning, demonstrating the ability to design, implement, and critically evaluate advanced deep learning models within a research oriented framework.

1.3 Goals

The goal of this project is to synthesize realistic FEM using state-of-the-art deep generative models. To achieve this, the work is divided into the following specific sub-goals:

- Reimplement and adapt existing scanpath generation models (DiffEyeSyn and SP-EyeGAN) for the task of unconditional FEM generation.
- Implement a Flow-based generative model, specifically the Rectified Flow architecture, for FEM synthesis, building on recent advances in Flow Matching approaches applied in other domains.
- Compare the performance of the three generative approaches using the JSD, evaluated across two different data preprocessing pipelines.
- Assess the realism of the generated data through quantitative analysis, including the examination of microsaccade characteristics in the synthetic FEM samples.

1.4 Research Methodology

This research investigates the comparative performance of deep generative models for FEM data synthesis. It adopts an experimental and quantitative research approach grounded in comparative model evaluation. The project focuses on assessing how different deep generative architectures such as

GANs, Diffusion Models and Rectified Flow Models perform in synthesizing realistic FEM data.

To ensure a fair and unbiased comparison, all models are implemented with comparable parameter counts and training conditions, so that performance differences can be attributed primarily to architectural design rather than model capacity. Each model is trained on the same FEM datasets under two distinct preprocessing pipelines: one inspired by DiffEyeSyn Sinusoidal Normalization and a Z–score Normalization.

Quantitative evaluation is performed using the Jensen–Shannon Divergence (2.2), a symmetric and information theoretic measure of distributional similarity that quantifies how distinguishable two probability distributions are from their average. This metric is widely used in prior generative modeling research and is well suited for assessing the alignment between synthetic and real FEM distributions.

In addition to quantitative analysis, qualitative evaluation is conducted through examination of microsaccade characteristics in the generated FEM data. These include the microsaccade rate, amplitude distribution, and main sequence relationship that are standard within the eye-tracking research community and serve as indicators of biological plausibility and realism in gaze behavior synthesis. Ensuring that the generated FEM data exhibit realistic microsaccade dynamics is particularly important for downstream clinical applications, such as early Parkinson's disease detection, where subtle irregularities in microsaccade dynamics may serve as diagnostic biomarkers.

The selection of the generative model families is motivated by both scientific relevance and novelty. GANs and diffusion models represent established paradigms previously applied in related eye tracking tasks, while Flow Matching models, particularly Rectified Flow remain underexplored and thus present a promising direction for investigation. Alternative approaches, such as Variational Autoencoders, were not considered in this project, as recent advancements have demonstrated that GANs, Diffusion and Flow-based methods generally achieve superior generative fidelity and sample diversity in comparable tasks.

1.5 Delimitations

Our limitations include the unconditional generation of FEM and does not include scanpath synthesis or stimulus-conditioned gaze modeling. Any

downstream clinical applications, such as diagnostic or classification tasks for Parkinson's disease, are also excluded from the scope of this work.

Furthermore, the study does not aim to conduct extensive hyperparameter tuning or a comprehensive investigation of architectural design choices within the evaluated models. Factors such as the number of layers, activation functions, or other implementation-specific variations are kept consistent across experiments to maintain comparability, rather than to optimize individual model performance.

1.6 Structure of the thesis

This manuscript is structured as follows. In Chapter 1, an introduction to the research background is given following with problem formulation including the purpose, methodology, and scope of the thesis. Chapter 2 presents the necessary theoretical background on FEM, distribution similarity metric, generative modeling approaches, and finally related work. Chapter 3 describes the research methodology, data selection, and experimental design. Chapter 4 details the proposed methods and implementation, including preprocessing, model architectures, and training procedures. Chapter 5 reports and analyzes the experimental results using quantitative and qualitative evaluation metrics. Chapter 6 discusses the findings, compares the evaluated models, and interprets their implications. Finally, Chapter 7 concludes the thesis and outlines limitations and directions for future work.

Chapter 2

Background

This chapter presents the theoretical and background knowledge necessary to understand the methods used in this thesis. It begins with a brief overview of FEM and their key characteristics, followed by detailed descriptions of the three generative model families examined in this work. Finally, the chapter reviews relevant prior research on eye-movement generation as well as recent advances demonstrating the effectiveness of Flow Models in time-series modeling.

2.1 Eye Movements During Fixation & Microsaccades

Our eyes are in constant motion, even during periods of fixation when we perceive our gaze to be stable. Fixational eye movements are typically decomposed into tremor, drift, and microsaccades [8]. Tremor corresponds to a high-frequency (30–100 Hz), extremely low-amplitude oscillation that often lies near the resolution limit of most eye-tracking devices. Drift refers to the slow, seemingly random movement of gaze between saccades, generally occurring at velocities below approximately $1°/s$ and producing cumulative displacements on the order of tens of arcminutes.

Microsaccades constitute the third component of fixational eye movement. They are small, jerk-like movements that resemble miniature saccades: binocular, ballistic, and producing the largest instantaneous gaze displacements during fixation. In the context of generative modeling, the realistic synthesis of

microsaccades is of particular importance, as these events reflect fine-grained oculomotor behavior relevant for downstream clinical applications.

Several characteristic properties of microsaccades have been described in the literature. They occur at an average rate of approximately 2 events per second, depending on the stimulus and task conditions [9]. Microsaccade amplitudes typically range from 1–25 arcmin (0.02–0.4°), with many studies adopting an operational definition of amplitudes below 1° of visual angle. The amplitude distribution is strongly right-skewed, characterized by a high frequency of very small events and a rapidly decreasing probability of larger amplitudes.

Another important feature is the main sequence relationship, which describes the relationship between saccade amplitude and peak velocity. When plotting peak velocity against amplitude for thousands of detected events, microsaccades form a near-linear trend. For healthy subjects, microsaccades with amplitudes between 0.1–0.5° typically reach peak velocities of around $120°/s$, and their peak velocity scales approximately linearly with amplitude, in agreement with main-sequence behavior reported for larger saccades.

2.2 Jensen Shannon Divergence

To compare data distribution of generated FEM versus real fixation, we utilize the JSD [10]. The Jensen Shannon Divergence is a symmetric and bounded dissimilarity between two probability distributions. It is formally defined as:

$$JSD(P||Q) = \frac{1}{2}KL(P||M) + \frac{1}{2}KL(Q||M), \ \ where \ \ M = \frac{1}{2}(P+Q), \quad (2.1)$$

where $KL(* \ || \ *)$ is the Kullback-Leibler divergence, defined as:

$$KL(P||Q) = \sum_{x \in X} P(x) \log(\frac{P(x)}{Q(x)}) \quad (2.2)$$

The Kullback Leiber divergence is known as relative entropy, which measures how the two probability distributions differ from each other. Unlike Kullback Leiber divergence, JSD is always finite and symmetric and takes values between [0, log2]. A value of 0 indicates identical distributions and larger values indicate increasing mismatch.

Since generative modeling typically aims to approximate or learn data

distribution $p(x) \approx p_{data}(x)$, JSD can be used to evaluate how well synthetic data resembles real data. A low JSD indicates that the generated FEM matches the real distribution and a higher JSD suggests that the generative model does not consistently capture FEM dynamics. Overall JSD provides a quantitative score summarizing how well the model produces synthetic data.

2.3 Generative Deep Learning Models

Generative models aim to learn the underlying data distribution $p(x)$ such that newly generated samples $\tilde{x} \sim p(x)$ resemble real samples. Depending on their formulation, some generative models learn an explicit representation of the data distribution, while others approximate it implicitly through sampling or adversarial training. The following sections describe the three model families examined in this work in detail.

2.3.1 Diffusion Models

Diffusion models [3] learn to reverse a gradual noising process by estimating the score, the gradient of the log-density of the noised data distribution at multiple noise levels. During sampling, these score estimates (or equivalently, denoising predictions) are used to iteratively move a noisy sample toward regions of higher probability mass, ultimately producing high-fidelity outputs.

The forward diffusion process is defined as a pre-specified Markov chain that adds small amounts of Gaussian noise to the data over T time steps:

$$x_t = \sqrt{\alpha_t x_0} + \sqrt{1 - \alpha_t \epsilon} \quad \epsilon \in N(0, I) \tag{2.3}$$

where t controls how much of the original data remains at step t. As t increases, the sample becomes progressively more corrupted until it approaches pure Gaussian noise. The model is trained to learn the reverse diffusion process, which conceptually corresponds to estimating the distribution $p_\theta(x_{t-1} \mid x_t)$.

In practice, reverse diffusion is implemented by training a neural network to predict either the added noise or the denoised sample x_0 at each time step. Denoising Diffusion Probabilistic Models (DDPM) for example, learn this reverse chain by minimizing the sum of squared residual loss between added

noise and predicted noise:

$$L = \mathbb{E}_{x_0,t,\epsilon}[||\epsilon - \epsilon_\theta(x_t, t)||^2] \tag{2.4}$$

During sampling, the process begins from pure Gaussian noise. The model then applies its learned denoising step iteratively, removing a small amount of noise at each iteration until a clean synthetic sample is obtained.

Subsequent research introduced variants such as Denoising Diffusion Implicit Models DDIM [11], which enable deterministic sampling and significantly reduce sampling time. While DDPM define a stochastic Markovian reverse process, DDIM replaces this with a non-Markovian deterministic transition: the next sample in the reverse chain is computed directly from the model's predicted noise, without the addition of stochastic noise.

2.3.2 Generative Adversarial Network Models

GANs operate through an adversarial training framework involving two neural networks: a Generator G and a Discriminator D [12]. GANs were among the first deep generative models capable of producing sharp, high-fidelity samples, particularly in image generation.

The core idea is that the generator learns to transform random noise, typically sampled from a Gaussian distribution, into synthetic samples that resemble real data, while the discriminator attempts to distinguish between real samples and those produced by the generator. Training proceeds as a minimax game, in which the discriminator seeks to maximize its classification accuracy, whereas the generator seeks to minimize the discriminator's ability to detect synthetic outputs. The original adversarial learning objective is defined as:

$$min_G \ max_D \ \mathbb{E}_{x\sim p_{data}}[\log D(x)] + \mathbb{E}_{z\sim p(z)}[\log(1 - D((G(z)))] \tag{2.5}$$

Training typically alternates between two steps:

- Updating the discriminator parameters to improve its ability to classify real versus synthetic samples.
- Updating the generator parameters so that the generated samples are more likely to be classified as real by the discriminator.

Substantial architectural improvements have been proposed since the original GAN formulation, with one of the most influential being the Deep

Convolutional Generative Adversarial Networks (DCGANs) [13]. DCGANs introduced the use of convolutional neural networks rather than fully connected layers, leading to more stable training and significantly higher-quality outputs. Instabilities commonly observed in early GANs, such as mode collapse and gradient vanishing were mitigated through the use of convolutional layers, batch normalization, and Rectified Linear Unit (ReLU) activation functions. These design principles have since become standard across many modern GANs architectures.

2.3.3 Flow Models & Rectified Flow

Flow-based generative models aim to generate data by learning a continuous-time transformation that transports samples from a simple base distribution (e.g., a Gaussian) to the target data distribution. Two recent approaches in this family are Flow Matching [14] and its improvement Rectified Flow [15], both of which learn a velocity field that defines how samples evolve over time according to an Ordinary Differential Equation (ODE):

$$\frac{dx_t}{dt} = v_\theta(x_t, t) \tag{2.6}$$

2.3.3.1 Flow Matching

Flow Matching trains a neural network to predict the velocity field that connects samples from an initial distribution π_0 (e.g., Gaussian noise) to samples from the target distribution π_1 (the data). This is accomplished by sampling paired points (x_0, x_1) from a coupling of the two distributions i.e., a joint distribution whose marginals are $x_0 \sim \pi_0$ and $x_1 \sim \pi_1$.

A common choice is to draw these pairs independently which results in a random coupling. Given a pair (x_0, x_1), Flow Matching uses a linear interpolation path:

$$x_t = (1 - t)x_0 + tx_1 \tag{2.7}$$

whose true velocity is simply:

$$\frac{dx_t}{dt} = x_1 - x_0 \tag{2.8}$$

The model is trained to match this velocity using an sum of squared residuals

objective:

$$L_{FM} = \mathbb{E}_{(x0,x1),t}[\|v(x_t, t) - (x_1 - x_0)\|^2] \tag{2.9}$$

After training, samples are generated by solving the ODE forward in time, transporting noise samples toward the data distribution. However, because the coupling between x_0 and x_1 is random and changes across mini-batches, the learned flow may not produce globally straight trajectories. The velocity field becomes an average of many possible directions, which can result in curved or inefficient transport paths.

2.3.3.2 Rectified Flow

Rectified Flow builds directly on the Flow Matching framework but introduces a crucial modification to the coupling strategy. Instead of using arbitrary data noise pairings, RF constructs a deterministic coupling between the distributions. The difference lies in how the couplings are chosen and updated. Rectified Flow enforces a consistent mapping between noise samples and data samples, rather than pairing them randomly. This ensures that:

- Each noise sample is consistently transported toward one specific data point.
- The model learns a stable, non-conflicting velocity field.
- The resulting ODE trajectories follow straight or nearly straight paths toward the data distribution.

Rectified Flow further strengthens this behavior through a reflow process:

1. Train a flow using the Flow Matching objective and deterministic coupling.
2. Generate new sample pairs by integrating the learned flow.
3. Treat these generated pairs as the new coupling.
4. Retrain the flow on the updated coupling.
5. Repeat.

Each iteration reduces curvature in the learned trajectories, producing progressively straighter transport paths. By enforcing a consistent pairing and refining it through reflow, Rectified Flow effectively solves a near-optimal transport problem under a convex cost. The straight-line trajectories emerge not from changing the interpolation or objective, but from rectifying the

geometry of the transport coupling itself. Because straight ODE trajectories can be simulated accurately with very few integration steps, Rectified Flow often produces faster sampling and high-quality results with significantly fewer computations than diffusion models.

2.4 Related Work

Previous research in the field of eye tracking has mainly focused on scanpath generation using state of the art models such as Diffusion and GANs. Therefore in the following an introduction is given to this prior work which will then serve as inspiration to the model used in this work. Additionally the novel application of Flow Matching in related domains is also explored.

2.4.1 DiffEyeSyn

DiffEyeSyn [7] introduced user specific high frequency eye movement synthesis. The motivation was based on eye movement biometric research showing that high frequency eye movements contain strong user specific signatures which are essential for identification models. The paper formulates user specific eye movement synthesis as the task of injecting user specific information back into a given eye movement sequence. Thus interpreting the user identity as a form of high frequency noise embedded in the raw velocities. The goal is therefore to learn a generative model capable of restoring these user specific dynamics from identity removed gaze sequence. This could be then used for downstream task of generating large amounts of personalized training data for biometric models.

The generative model used in the paper was a conditional DDPM, trained to reconstruct the original high frequency velocities, conditioned on identity removed velocity. The DiffEyesSyn is built on the DiffWave [16] architecture, which was originally for high frequency audio. This was motivated with the use of bidirectional dilated convolutions to efficiently handle very long sequences such as 1000 Hz eye movement.

In addition to the standard DDPM noise prediction loss, the paper introduces a novel User Identity Guidance Loss: after predicting the denoised sample $\tilde{x}$, compute the cosine similarity between embedding of sample $\tilde{x}$ and true x. With the loss maximizing the similarity to ensure the generated sequence carries the target user identity.

In their results, for Gazebase dataset, DiffEyeSyn shows a within user similarity score of 0.787 compared to human baseline of 0.498, and without the identify guidance loss the performance drops to 0.322 which highlighting that the model does not consistently add any meaningful user specific information. Additionally, they compared the velocity distribution using JSD metric, where DiffEyesyn synthetic gaze has an overall $JSD = 0.0035$ compared to human baseline of 0.0389 which suggests synthetic velocities are more similar to real human velocities than two real humans are similar to each other. In essence DiffEyeSyn showcases strong reconstruction of user identity using conditional DDPM built on DiffWave architecture.

2.4.2 SP-EyeGAN

SP-EyeGAN [6] proposed a GANs based framework for generation of synthetic raw eye tracking data with realistic fixation and saccades, primarily for reading context. The motivation was that high quality raw eye tracking dataset are expensive to collect and raise privacy concerns and deep models that operate directly on raw gaze signals need large amounts of data to avoid overfitting. Therefore, SP-EyeGAN would address this by generating large amounts of synthetic sequences for downstream tasks.

Sp-EyeGAN decomposes eye movements into saparetes GANs, FixGAN: generate short segments of fixational eye movement and SacGAN: generate short segments of saccadic movement, both trained on respective data. Both GANs operate on velocity data extracted from GazeBase poem reading task, and fixations and saccades were separated using Dispersion-Thershold Identification algorithm. The architecture consists of dilated convolutional residual blocks, where each generator maps a noise vector through a fully connected layer and three deconvolutional blocks to produce a short velocity sequence, 100 *ms* for fixations and 30 *ms* for saccades. The corresponding discriminator is 1D convoulitional neural network with tree convolutional blocks and a final sigmoid output that classifies real versus synthetic segments.

To construct scanpaths, SP-EyeGAN relies on external fixation location and duration models. Given a sequence of fixation positions and sampled duration, the model alternates between sampling a fixation sequence from FixGAN and sampling a saccade sequence from SacGAN to which rotates it so that it lands on the next fixation location. This workflow yields a continuous synthetic raw gaze sequence.

Beyond generation of scanpath, the paper proposes a contrastive pre-training

framework where SP-EyeGAN is used to produce large amounts of synthetic raw eye movement sequences, which then serve as unlabeled data for contrastive learning of deep encoder such as CLRGaze and EKYT. Moving on to results, Sp-Eye GAN showcases that SP-EyeGAN achieves synthetic data closer to human data than prior statistical and Variational Autoencoder based generators. This is done via JSD metrics, where for synthetic fixation SP-EyeGAN $JSD = 0.029$ compared to EyeSyn $JSD = 0.064$ and Variational Autoencoder $JSD = 0.201$. However as part of papers limitation, SP-EyeGAN does not generate fixation locations or durations itself but demonstrates that GANs based generators can produce high quality fixation and saccade eye movements. Furthermore, the paper also does not validate the characterticis of generated fixation from FixGAN.

2.4.3 ScanDL

ScanDL [17] aimed to introduce a generative model to synthetic scanpath from a given sentence as input. ScanDL 2.0 [18] extends the original ScanDL framework which is a discrete sequence to sequence diffusion model for predicting fixation location, by introducing a new module for predicting both fixation location and durations. The motivation of the paper states that previous models including GANs and RNNs failed to model the dual sequence structure of reading (linguistic order vs fixation order) and none could synthetize fixation durations, despite the crucial importance of duration for reading behavior.

The model overview of ScanDL 2.0 consists of two complementary components: ScanDL module which is a discrete diffusion model that predicts fixation locations by jointly embedding sentence tokens and fixation index sequence (i.e. scanpath). It applies partial noising to only the scanpath portion during the forward diffusion process and uses an encoder-only Transformer to perform reverse denoising. The second component is the fixation duration module which is a Transformer based sequence to sequence model conditioned on the predicted fixation locations. It is trained separately with the mean squared error loss on scaled durations.

Within dataset performance, ScanDL 2.0 shows substantial improvement over previous computational cognitive models and thus represents one of the most comprehensive generative models of reading behavior.

2.4.4 FlowTS

FlowTS [19] is a rectified flow based generative model for unconditional as well as conditional time series generation. Although it is outside the eye tracking domain it demonstrates strong performances in comparison to diffusion models for sequential data. The main motivation of the paper is to address Diffusion models computationally expensive inference. Rectified Flow model aims to serve as an efficient, stable and high fidelity alternative to diffusion models. This is done by replacing the denoising trajectory of diffusion with a straight line transport between a Gaussian prior and the data distribution. In doing so, FlowTS drastically reduces sampling steps, eliminates numerical instability from iterative solver whilst generating high quality synthetic time series data.

FlowTS is an ODE based generative model that uses Rectified Flow to learn a deterministic transport map from noise to real data samples. Instead of learning a noisy stochastic reverse process like diffusion models, instead the model learns the velocity field of a linear path between these distributions. To further improve quality of generated samples, FlowTS introduces adaptive time stepping:

$$t_{i+1} = (\frac{i+1}{N})^k,\ k \in (0,1] \tag{2.10}$$

ti+1=(i+1N)k, k (0,1].

The adaptive time stepping means that early steps resemble coarse exploration, where later steps are fine exploitation with the aim of improving efficiency and quality at the same time. Overall, the architecture used in FlowTS is a Transformer encoder-decoder backbone.

The dataset used in FlowTS is financial time series as well as energy consumption for which they employed components such as Trend & Seasonality Decomposition, attention register plus Rotary Position Embeddings in the architecture to help the model capture both local and global structure. As a result, FlowTS demonstrates that rectified flow models can outperform diffusion-based approaches for time-series generation while being significantly faster at inference. Across six benchmark datasets, FlowTS achieves state of the art performance in unconditional generation, showing consistently lower Context-FID, discriminative scores, and correlation errors than diffusion and GANs based baselines.

2.4.5 Trajectory Flow Matching

Trajectory Flow Matching [5] is a recent proposed generative modeling designed specifically for stochastic and irregularly sampled time series data with a primary focus on clinical datasets. It is stated that traditional Neural Stochastic Differential Equation (SDE) provide a natural way to model uncertainty and continuous time dynamics but they require computationally expensive backpropagation through SDE solvers, which limits scalability and can lead to training instability. Thus the paper aims to address this by introducing a simulation free training method based on Flow Matching which enables faster and more stable learning of continuous stochastic dynamics. This is especially useful in healthcare applications where real world data is noisy and irregularly sampled.

Trajectory Flow Matching is motivated by the need for a method that learns continuous time dynamics without expensive solver based training, handles stochastic by learning both drift and diffusion and supports irregular sampling and conditional modeling whilst also scaling efficiently for large time series datasets. To achieve this, it adapts the Flow Matching to time series modeling by directly learning the instantaneous velocity and uncertainty of time dynamic without differentiating through any differential equation solver.

Trajectory Flow Matching constructs local conditional bridges between consecutive time points. For each pair of observations (x_t, x_{t+1}), it samples an intermediate time $t + \tau$ and fits a velocity field and a noise uncertainty predictor. This allows the model to approximate a Neural SDE without numerical simulation during training. Additionally it is important to ensure the predicted trajectories respect the true pairing structure of the data, Trajectory Flow Matching proves conditions under which coupling is preserved and in practice enforces this by conditioning the flow on a history window, meaning previous observations. Instead of predicting velocity directly, Trajectory Flow Matching can precut the future observation x_{t+1} and convert this into a velocity. Trajectory Flow Matching also learns, x_{next} which represent the next observation value and tnext the time of the next measurement. Trajectory Flow Matching was evaluated on four challenging clinical time series datasets and shows substantial improvement in prediction accuracy and strong uncertainty estimation.

2.5 Summary

In summary, this chapter introduces the theoretical background required for the thesis. FEM including tremor, drift, and microsaccades have been extensively studied, highlighting their dynamics and relevance for generative modeling. The thesis introduced three families of generative models, Diffusion, GANs and Flow based models suited for generating synthetic FEM.

Diffusion learns to reverse a noise adding process by predicting noise at multiple timesteps. GANs use an adversarial framework between generator and discriminator to learn synthesized real data. Flow based models, more specifically Flow Matching learn to fit velocities along a linear interpolation and Rectified Flow imposes deterministic pairings and iterative reflow to straighten trajectories for faster inference.

Prior work on eye movement generation and related models were also explored, offering an insight to how these families of generative modeling have been utilized to model eye movement data. In DiffEyeSyn, a conditional DDPM built on Diffwave demonstrates the ability to restore user specific high frequency gaze identity. Whilst, SP-EyeGAN trains two separate GANs to generate both fixations and saccade to produce realistic short sequences of eye movement. ScanDL 2.0 showcases the capabilities of a discrete diffusion model predicting fixation locations and durations, offering a strong performance in reading tasks compared to previous models. Related works in Flow base approaches were also explored, such as FlowTS using Rectified Flow model plus adaptive time stepping to produce high quality samples.

These developments in the eye tracking domain highlight the use of generative modeling and its application in eye movement synthesis. Yet despite the extensive work on scanpaths, saccades, reading behavior or biometrics, no existing approach has purposefully attempted to model FEM. Continuing forward, this thesis addressed this gap.

Chapter 3

Methodology

In this chapter, the methodological framework used in this thesis is outlined, covering the research process, chosen paradigm, data selection and experimental setup used to compare the three selected generative models for FEM synthesis.

3.1 Research Process

The research process began with a literature review on existing work on generative modeling within the eye tracking domain. The literature review, dataset and preprocessing conventions as well as architectural choices were examined. This review provided the foundation for identifying the research gap and established the experimental setup for a systematic comparison between Diffusion, GANs and Rectified Flow and informed the choices of dataset, evaluation metrics used in this project.

Following the literature review, the generative models were adapted to FEM generation tasks. The engineering workflow consisted of designing comparable architectures, establishing consistent training conditions and implementing a controlled evaluation setup suitable for direct comparison. The workflow reflected an experimental method in which the hypothesis about model results were tested through structured evaluation that are aligned with prior work.

The process concluded with the analysis and interpretation of the generated FEM, allowing the results to be connected back to the stated research goals. Thus, the research process of this thesis ensures that the final conclusions are

methodologically grounded and reproducible and aligned with the aims of this project.

3.2 Research Paradigm

This project follows a quantitative and empirical research paradigm. The results of this project are numerical evaluation of model outputs, where hypotheses about generative model performance are tested through measurable metrics such as distribution similarity and spectral characteristics as well as physiological features of FEM. The conclusions are based on reproducible experiments conducted under controlled conditions therefore making this paradigm appropriate for a comparative analysis of generative models. In contrast, similar research on generative models for image or audio synthesis may adopt a qualitative approach which is not suited for this thesis. This is because the project does not involve subjective evaluation but rather grounds results in an empirical validation to assess the performance across the models.

Additionally, the project adopts an engineering design paradigm since the core contribution of the thesis is the addition of a novel model Rectified Flow for FEM synthesis as well as the evaluation of two other generative models adapted for FEM synthesis. Therefore the research process involves designing model architecture and implementing a training pipeline and assessing their performance.

3.3 Data Collection

For this project, the Gazebase dataset [20] was used. Gazebase is a publicly available eye tracking corpus that includes multiple standardized recording tasks. As this work focuses on FEM synthesis, only the Fixation task was used, which consists of 15 seconds recording where participants maintain fixation on a central dot. The recording is well suited for modeling natural fixational behavior during a simple fixation task, where visual input and instructions are kept minimal and consistent across trials.

Stimulus driven tasks within Gazebase were excluded because they are not pure FEM sequences which lies outside the scope of this work. scanpath datasets were not suitable due their inclusion of saccades and task dependent patterns rather than pure fixation. Another candidate dataset was Judo1000 [21], a dataset that almost resembles the fixation task because the participants

are instructed to fixate on the central dot. However, in Judo1000 the fixation target jumps to a new location every few seconds which induces task driven saccades. Although fixation segments can be extracted using fixation detection algorithms, as done in prior work, this approach results in short, concatenated fixation snippets rather than continuous long duration fixation. This approach is not ideal because the length of each sustained fixation would be too short compared to the window length considered in this work. Therefore, Judo1000 was not selected for this project.

From an ethical standpoint, GazeBase is anonymized and ensures that no personal or identifying information is used, thereby supporting responsible data-use practices. Furthermore, Gazebase was released with the intent of being used in research and therefore was chosen as a suitable dataset for this project.

3.4 Experimental Design

The experimental setup of this work is designed to evaluate how three generative model families, Diffusion, GANs and Rectified Flow model perform in the task of unconditional FEM generation. The central objective is to establish a controlled comparison in which any observed difference in performance can be attributed to the generative paradigm rather than external factors.

The independent variable in this experiment is the choice of generative model and data preprocessing. Between all models, the Rectified Flow and Diffusion model share similar roots in architecture and therefore to ensure comparability between them, a similar backbone architecture is used with similar parameter count. The GANs however, is larger in parameter count because of the necessity of a larger projection layer.

The experimental procedure consists of training each model on the same dataset, using the two proposed data preprocessing methods, then generating a fixed number of synthetic FEM sequences and evaluating the outputs using complementary metrics. The evaluation framework includes: basic statistical comparisons (mean, standard deviation, maximum value per channel), distribution similarity assessed through Jensen Shannon Divergence (including a real to real baseline between subjects), frequency domain plot of average Power Spectral Density and also a visual inspection of generated sequences. To further assess the realism of the generated sequence,

microsaccades are extracted from both real and synthetic data and the microsaccade rate and main sequence relationships are compared.

3.5 Evaluation of Method & Framework

The described evaluation framework aims to ensure that the results obtained in this project are reliable and valid. Reliability is addressed by maintaining a controlled evaluation condition across all models, including preprocessing to reduce variability. Also, all metrics are computed over multiple generated samples to reduce uncertainty, and also a real to real JSD baselines is used to provide a stable reference point against which model performance can be interpreted against. Validity is supported by the use of common metrics that have been used in prior works in the eye tracking domain as well as directly reflect the project goal of FEM synthesis which mainly includes the realism of microsaccade characteristics.

The evaluation methodology has some limitations. For instance the JSD depends on histogram binning choices and microsaccade characteristic evaluation can be affected by the choice of extraction algorithm. Future work could incorporate downstream task performance, such as the one described as the broader goal of this project, by evaluating if the generated data can augment diagnostics models. Nevertheless, the chosen method for this project provides a reliable and valid basis for comparing generative model families in the context of FEM synthesis.

Chapter 4

Method and Implementation

In this chapter, we describe the full pipeline for generating FEM sequences. The workflow begins with a preprocessing applied to the GazeBase dataset using the two different normalization methods. Each model is trained separately on the two preprocessing methods. The selected models' architecture is outlined alongside the training setup and configuration and finally how each model generated FEM sequences. Together these components define the method used whose output is evaluated in Results Chapter 5.

4.1 Data Preprocessing

The Gazebase dataset was loaded using the PyMovement [22] library and only the Fixation Task recording was extracted. This task consists of about 15 seconds of continuous fixation on static target. Each recording contains horizontal x and vertical y gaze coordinates sampled at 1000 Hz. All gaze positions were centered by subtracting the target dot location from the recorded coordinates.

Blinks were removed and remaining short gaps were linearly interpolated. The blinks were identified as any continuous NaN segments of 200 *ms*. Also, a 50 *ms* buffer was removed before and after each blink to eliminate partial closures, furthermore overlapping intervals were merged. All samples within this interval were thus discarded. After blinks removal, shorter segments were processed using linear interpolation. This step was necessary to ensure temporal continuity, as the generative models assume continuous sampled signals. Any remaining NaNs after the interpolation step were simply dropped.

The remaining sequence was smoothed using a convolution based Bartlett window filter using a window length of 23 samples. This was done to suppress high frequency sensor noise while preserving microsaccade and drift components. After which the gaze coordinates were converted to velocity using the central difference method. The interior points use the 5 point stencil and boundary points use a 3 point approximation. This method for velocity calculation was chosen because of its numerically stable gradient estimates. After computing velocity, values were clipped to the range [-200, 200] to remove extreme spikes to prevent outliers from dominating the signal. This is the common preprocessing for the following two methods used in this project, after which two different approaches to normalization were considered.

4.1.1 Sinusoidal Normalization

The sinusoidal normalization was applied as proposed in DiffEyeSyn. The velocity values were first scaled by dividing by $200°/s$ and then transformed using a sine function:

$$v_{sin} = sin(\frac{\pi}{2}\frac{v}{200}) \tag{4.1}$$

This transformation smoothly compressed large velocities while preserving the relative structure of smaller fluctuations. The motivation follows from DiffEyeSyn that this method retrain microsaccade characteristics more effectively than Z-score Normalization.

4.1.2 Z-score Normalization

Z-score normalization was implemented as common data preprocessing step for generative modeling, defined as:

$$v_{z-score} = \frac{v - \mu_{global}}{\sigma_{global}} \tag{4.2}$$

Unlike per recording normalization, global mean and standard deviation were computed across all recordings in the dataset. The preprocessing pipeline there used two passed:

- A first pass to compute global statistics from all recordings.
- A second pass to normalize each sequence using the same global parameters.

Global z-score normalization ensures that the model receives inputs on a consistent scale across all training samples which is important for training stability.

4.1.3 Temporal Segmentation

After preprocessing and normalization, each fixation recording was segmented into fixed length windows of 5000 samples, which corresponds to the sequence length expected by the generative models. To achieve this, none overlapping segmentation was used where each recording was partitioned into consecutive windows of 5000 samples and any remaining partial segments were discarded. The final resulting training data after preprocessing was a uniformly sized tensor of shape (5000, 2) where each window represents a clean segment of horizontal and vertical velocity signal ready for model training.

4.2 Model Implementation

All models were implemented in PyTorch and developed using Jupyter notebooks. To ensure fair comparison across Diffusion and Rectified Flow models, they both used the same backbone architecture inspired by the DiffWave model used in DiffEyeSyn but without any conditioned input. The GANs model was adapted from FixGAN from SP-EyeGAN, which was originally designed for generating 100 *ms* fixation patches. To support the longer 5000 sample sequences the generator and discriminator were modified accordingly.

4.2.1 Diffusion

The diffusion model architecture is shown in Figure 4.1. It is based on the DiffWave backbone and implements a one-dimensional denoising network for velocity signals. The two-channel input velocity sequence is first projected into a 128-dimensional feature space using a 1D convolution. Temporal information is incorporated through a timestep embedding, implemented as a two-layer Multilayer Perceptron (MLP) with Sigmoid Linear Unit (SiLU) activations. The output of this embedding is added to the activations within each residual block, enabling the network to condition its predictions on the diffusion timestep.

The core of the network consists of 30 residual blocks, with a residual channel

of 128. Each block consists of 1D convolutions, skip connections and a repeating dilation schedule. All residual outputs are combined and passed through a skip output module to predict the 2-channel noise estimate. The architecture is fully convolutional and contains approximately 4.3 million parameters.

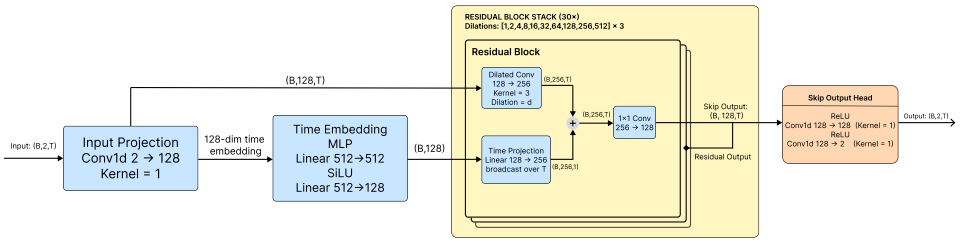


Figure 4.1: Diffwave architecture abstraction.

4.2.2 GAN

To support 5000 sample sequences the generator consists of a large input projection layer that takes a 256 dimensional latent vector and expands through a fully connected layer into an 80000 dimensional representation which then reshape into a 16 channel sequence. This layer accounts for the majority of the model's parameter count and is required to map the latent distribution into a high resolution temporal signal. The generator then applies a stack of 4 residual dilated blocks with kernel size 5 and dilations of 1, 2, 4 and 8. A final 1D convolution maps the 16 channels representation back to the two output channels.

The discriminator consists of a sequence of strided 1D convolution that progressively increases the channel width from 2 to 256 while reducing the temporal resolution. This is followed by a small high frequency convolutional module and a final linear layer that outputs a real or fake score. Leaky ReLU activation is used throughout the network. For overall view model architecture see figure 4.2.

In total, the GAN comprises approximately 21 million parameters with more than 20 million coming from the generator due to the large projection layer. This difference makes GAN significantly larger than the diffusion and Rectified Flow model but is unavoidable due to the architecture's reliance on signal latent to sequence step.

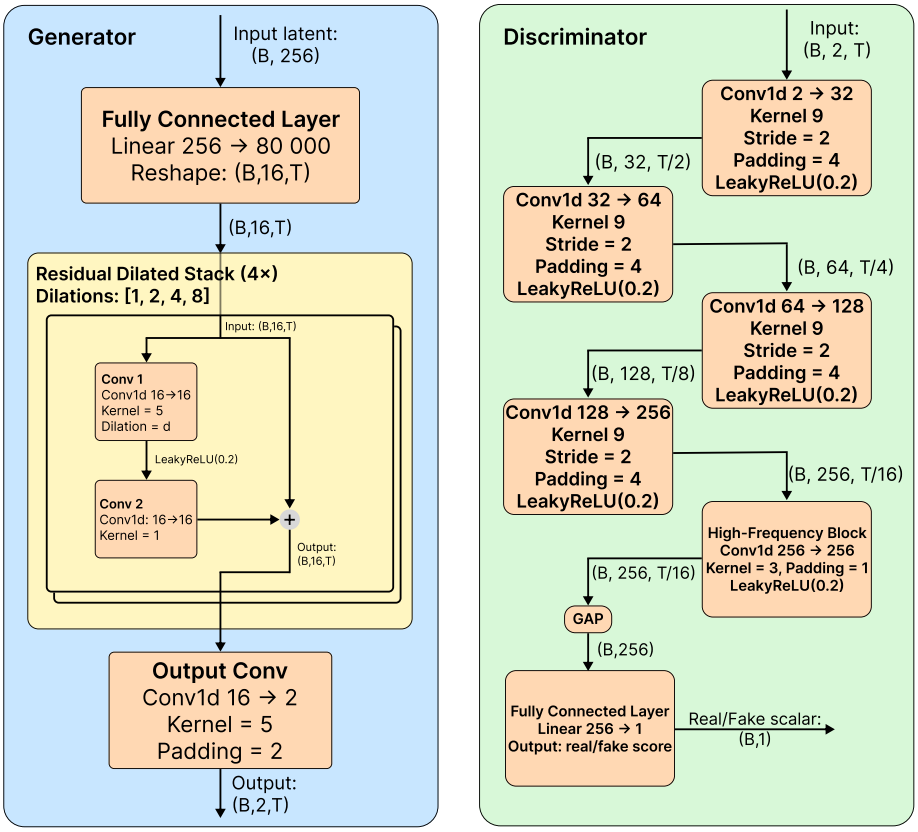


Figure 4.2: GAN model architecture of Generator and Discriminator abstraction.

4.2.3 Rectified Flow

The Rectified flow shares the same DiffWave backbone to the diffusion model therefore see architectural details in 4.2.1. In this framework, the model does not learn a denoising function but instead learns a time-dependent velocity field that transports samples from prior to target FEM distribution. As Rectified Flow is the novelty applied to FEM sequence synthesis in the thesis, the workflow from raw data to model training and inference is summarized in the following diagram, see figure 4.3.

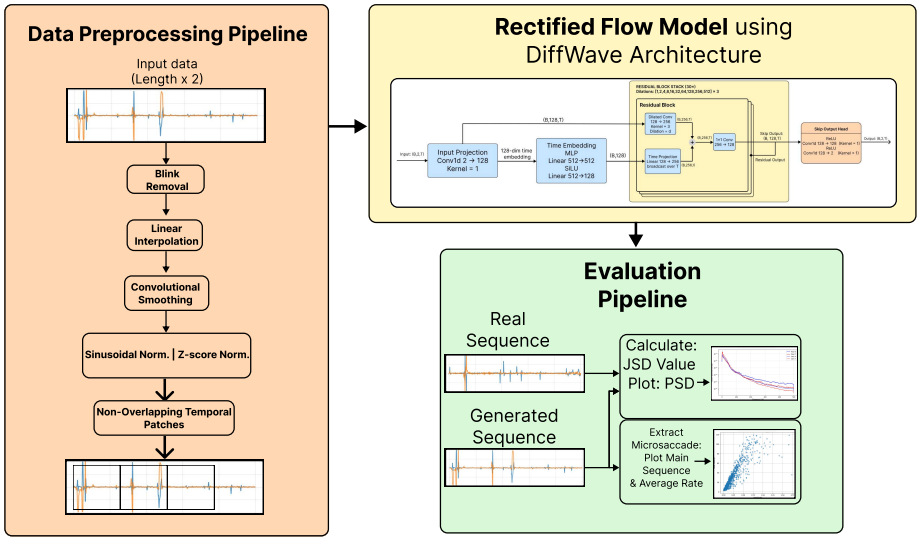


Figure 4.3: Full RF Pipeline from preprocessing to training and evaluation.

During training, the Rectified Flow model receives pairs of data samples and prior noise samples, from which intermediate states are constructed for learning the transport velocity field. Once the model has learned this field, inference proceeds by integrating it forward from Gaussian noise to generate complete FEM trajectories. More on the inference is explained in .

4.3 Training Setup

All models were trained on the two proposed data preprocessing methods using the same training configuration. For training, all of the data was used as training data, meaning there were no data split for test. This is because the objective is not to estimate out-of-sample predictive performance on a held-out test set, but to learn a generative model that closely matches the distribution of the data itself.

The hardware used for training was the Nvidia A100 provided from NAISS [23]. All models have PyTorch's default weight initialization. The training hyperparameter are shown in table 4.1.

The optimization settings were chosen to reflect the different training objectives of the generative models. In preliminary experiments, the diffusion and GANs models were also trained using AdamW with weight decay of

Table 4.1: Final Training configurations used for all models.

Setting	**Diffusion**	**GAN**	**Rectified Flow**
Optimizer	Adam	Adam	AdamW
Optimizer Betas	(0.9, 0.99)	(0.5, 0.9)	(0.9, 0.99)
Learning rate	1×10^{-4}	1×10^{-4}	3×10^{-4}
LR schedule	Warmup + cosine decay	Warmup + cosine decay	Cosine Annealing
Warmup steps	2000	2000	-
Final LR	1×10^{-6}	1×10^{-6}	-
Weight decay	–	–	0.01
Batch size	32	64	32
Epochs	300	1000	300

0.01, however this did not lead to improved training stability. In contrast, Rectified Flow training benefited from the use of AdamW and weight decay and therefore the configuration was chosen. Learning rates and learning rate schedules were chosen in line with what is commonly recommended for each generative model to ensure stable convergence [15] [24].

For Diffusion model, the training objective chosen is the DDPM noise prediction loss given as:

$$L = \mathbb{E}_{x_0,t,\epsilon}[||\epsilon - \epsilon_\theta(x_t, t)||^2] \tag{4.3}$$

with T = 50 diffusion steps and a linear schedule in range $[10^{-4}, 0.05]$.

For the GANs, the training objective chosen is the Wasserstein loss and gradient penalty given as:

$$L_D = \mathbb{E}[D(\tilde{x})] - \mathbb{E}[D(x)] + \lambda_{gp} GP, \;\; L_G = \mathbb{E}[D(\tilde{x})] \tag{4.4}$$

where λ_{gp}=10 and GP enforces the 1 Lipschitz condition.

For the Rectified Flow, the training objective chosen is the Flow Matching objective given as:

$$L = \mathbb{E}_{x_t,t,u}[||v(x_t, t) - u||^2] \tag{4.5}$$

where $u = x_1 - x_0$ and $x_t = (1-t)x_0 + tx_1$. The timestep sampling is inspired by FlowTS named Logit-Normal + Uniform Mix, where t is sampled from a

logit normal distribution producing value biased toward 0 and 1, mixed with a uniform sampling to avoid mode collapse at the edges.

4.4 Sample Generation

After training, the last checkpoint is loaded in to generate a FEM sequence using methods specified in subsections 4.4.1, 4.4.2 and 4.4.3, for each generative model the generated sample is saved as a numpy array to later be evaluated and compared to the real data.

4.4.1 Diffusion Sampling

The diffusion model follows the DDPM reverse process [3]. To generate a sample, the model begins with Gaussian noise $x_T \sim N(0, I)$ and the trained denoising network is then applied for 50 reverse timestep where at each step, the model predicts the noise component conditioned on the current timestep, and the schedule parameters determine how the signal is incrementally refined. This iterative process gradually transforms noise into FEM samples.

4.4.2 GAN Sampling

For the GAN, sampling is done with one forward pass in the generator. A latent vector is drawn from Gaussian distribution, $z \sim N(0, I)$ and passed directly through the generator $\tilde{x} = G(z)$ 1.

Algorithm 1: Sample Generation with a Generative Adversarial Network

[1] **Input:** Trained generator G **Output:** Generated sample $\tilde{x}$ Sample latent vector $z \sim \mathcal{N}(0, I)$ Generate sample $\tilde{x} \leftarrow G(z)$ **return** $\tilde{x}$

4.4.3 Rectified Flow Sampling

For Rectified Flow, two methods for sampling were proposed. Both methods start from Gaussian noise but differ in how they integrate the learned velocity field over time.

The Heun RK2 sampling is used [25], which is a midpoint ODE integrator. This method is deterministic and aims to approximate the continuous rectified flow ODE with improved stability compared to a simple Euler integration. Given a polynomial time grid: $0 = t_0 < t_1 < ... < t_K = 1$:

Algorithm 2: Rectified Flow Sampling using Heun RK2

Input: Initial sample $x_{t_0} \sim \mathcal{N}(0, I)$, trained velocity field v_θ, time grid $\{t_k\}_{k=0}^{K}$

Output: Generated sample x_{t_K}

for $k = 0$ **to** $K - 1$ **do**

Set step size $h = t_{k+1} - t_k$;

Predict velocity at current state:;

$v_1 \leftarrow v_\theta(x_{t_k}, t_k)$;

Compute midpoint state:;

$x_{\text{mid}} \leftarrow x_{t_k} + \frac{h}{2} v_1$;

Predict velocity at midpoint:;

$v_{\text{mid}} \leftarrow v_\theta(x_{\text{mid}}, t_k + \frac{h}{2})$;

Update sample:;

$x_{t_{k+1}} \leftarrow x_{t_k} + h\, v_{\text{mid}}$;

return x_{t_K}

4.5 Microsaccade Extraction

For evaluation of microsaccade characteristics, all generated samples were reverse normalized prior to microsaccade extraction. Microsaccades were identified using the Engbert & Kliegl detection algorithm [9]. Accordingly, each generated sequence was transformed back to the original signal scale using either Sinusoidal or Z-Score inverse normalization, after which microsaccades were extracted for subsequent analysis. For parameters, we used the velocity threshold of 5 and the minimum microsaccade duration was set to 3 samples. The microsaccade detection was done under identical pipelines for both real and generated.

Chapter 5

Results and Analysis

We present the results of the experimental evaluation of FEM sequence generation using three generative model families, diffusion, GANs and Rectified Flow are presented. The generated samples were compared against real FEM using multiple metrics, including basic velocity metrics, time series visualization, PSD, JSD and finally microsaccade characteristics which includes, main sequence analysis and microsaccade rate.

5.1 General Properties of Generated Fixations

Basic metrics were computed for both horizontal and vertical velocity components including mean, standard deviation and maximum absolute velocity. These metrics provide a initial indication of whether the generated sequences fall within ranges of the real data. Results for the real data and all models across both normalization methods are summarized in Table 5.1 and 5.1. The values highlighted in bold are the closest value to real data.

Rectified Flow exhibits comparable results to real data, particularly under Z-score normalization. In contrast, Diffusion generated sequences show significantly lower velocity variance and max absolute value, indicating smoothed sequence. However, under Sinusoidal normalization the diffusion generated sequence aligns more closely with real data, followed closely by Rectified Flow. By comparison, GANs generated sequences show consistently lower velocity variance in Z-score normalization as well as Sinusoidal Normalization, indicating less variability compared to real data. The main

Table 5.1: General metric comparison between real data and generated sequences from three generative models, trained using Z-score Normalization. Vx is horizontal velocity component and Vy is the vertical velocity component.

Metric	Real	Rectified Flow	Diffusion	GAN
Vx_{mean}	0.001	-0.004	-0.005	**0.000**
Vx_{std}	1.045	**0.863**	0.237	0.447
Vx_{max}	10.894	**11.666**	3.522	16.688
Vy_{mean}	-0.007	**-0.005**	-0.032	0.000
Vy_{std}	0.982	**0.925**	0.186	0.424
Vy_{max}	7.868	8.515	3.637	**7.823**

Table 5.2: General metric comparison between real data and generated sequences from three generative models, trained using Sinusoidal Normalization. Vx is horizontal velocity component and Vy is the vertical velocity component

Metric	Real	Rectified Flow	Diffusion	GAN
Vx_{mean}	0.001	0.0003	**0.0007**	0.000
Vx_{std}	0.109	0.095	**0.109**	0.046
Vx_{max}	1.000	**1.237**	1.396	1.342
Vy_{mean}	0.005	0.007	**0.004**	0.000
Vy_{std}	0.139	**0.137**	0.149	0.067
Vy_{max}	1.000	**1.146**	1.375	1.205

takeaway of these finding suggest that diffusion paired with Sinusoidal normalization better captures FEM dynamics at a statistical level whilst Rectified Flow shows comparable results using both normalization methods.

5.2 Distributional Similarity

Distributional similarity between real and generated sequences was quantified using JSD. Velocity magnitude distribution were estamited via histograms and JSD was computed between real and generated samples for all three generative models. As a reference, JSD was computed for real data for inter-subject variability. Results are summarized in Table 5.3 and 5.4. The values highlighted in bold are the best.

Table 5.3: JSD comparison between real data and generated sequences from three generative models, trained using Z-score Normalization.

Metric	Real	Rectified Flow	Diffusion	GAN
JSD_{mean}	0.089	**0.078**	0.120	0.201
JSD_{std}	0.044	**0.038**	0.064	0.048

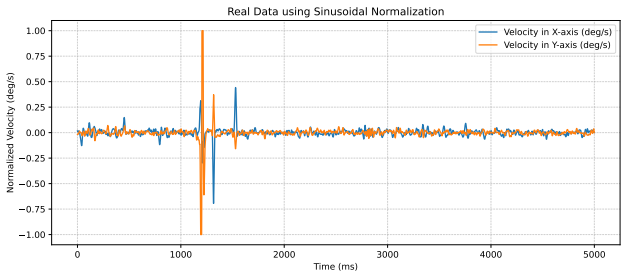
Table 5.4: JSD comparison between real data and generated sequences from three generative models, trained using Sinusoidal Normalization.

Metric	Real	Rectified Flow	Diffusion	GAN
JSD_{mean}	**0.107**	0.118	0.128	0.195
JSD_{std}	**0.051**	0.062	0.058	0.065

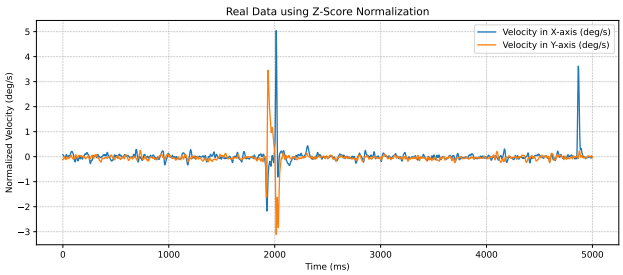
Under both normalizations, Rectified Flow achieves JSD values comparable to and in case of Z-score normalization lower than the inter subject baseline. The diffusion generated sequence across both normalization achieves JSD values slightly above the baseline range. Despite differences observed in statistical variability across normalization methods, the diffusion generated sequence shows similar levels of distributional similarity in both cases. In contrast, the GANs generated sequences has highest JSD values than other generative models. Among all evaluated models, the Rectified Flow model achieves the lowest distributional divergence under both normalization methods.

5.3 Visualized Plot of Generated Sequences & Real Data

In Figure 5.1, we present time series plot of real fixation signal under both normalization methods. Real fixation sequences displays low amplitude drift with sparse but brief velocity spikes corresponding to microsaccade. Similar pattern is observed in Figure 5.2, for Rectified Flow generated sequences. In Figure 5.2, for the diffusion model it becomes more clear that generated sequence using Z-score normalization results in less variability of true fixation dynamics and more noise. For GANs generated data in Figure 5.4, the sequence appears to have more noise and thus the temporal structure of the sharp characteristic of real fixation behavior are smoothed.

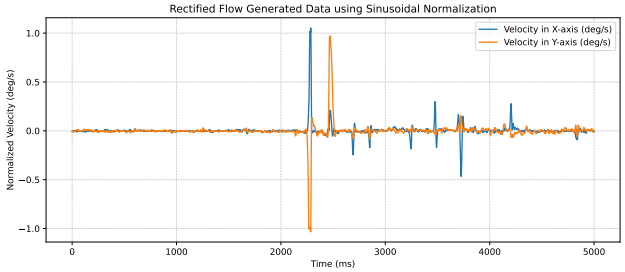


(a) Real FEM Sinusoidal Normalized Velocity.

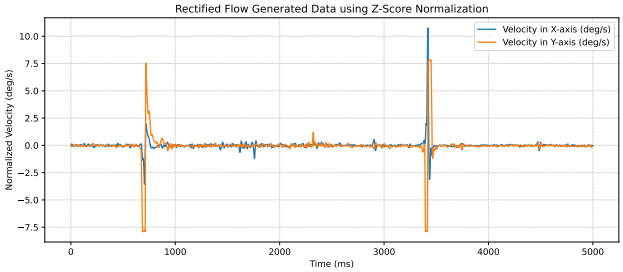


(b) Real FEM Z-Scored Normalized Velocity.

Figure 5.1: Comparison of real FEM velocity time series normalized with (a) sinusoidal normalization and (b) z-score normalization.

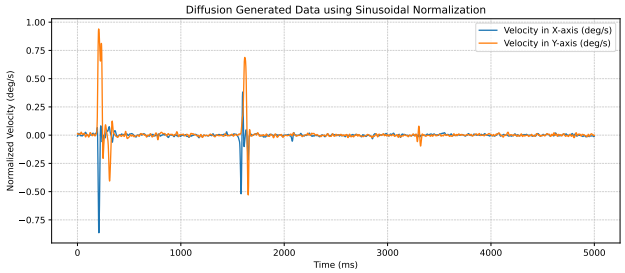


(a) Rectified Flow Synthesized FEM Sinusoidal Normalized Velocity.

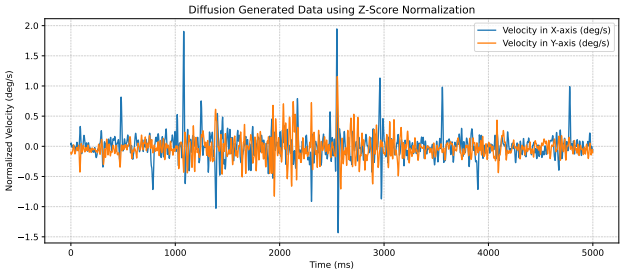


(b) Rectified Flow Synthesized FEM Z-Scored Normalized Velocity.

Figure 5.2: Synthesized FEM velocity time series from the Rectified Flow model, normalized with (a) sinusoidal normalization and (b) z-score normalization.

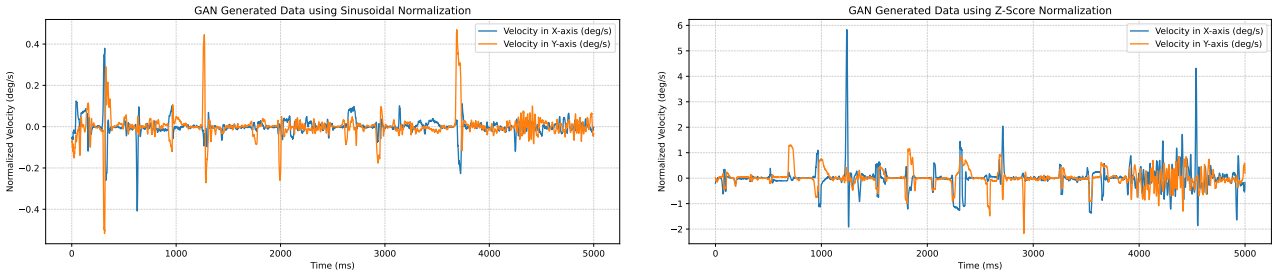


(a) Diffusion Synthesized FEM Sinusoidal Normalized Velocity.



(b) Diffusion Synthesized FEM Z-Scored Normalized Velocity.

Figure 5.3: Synthesized FEM velocity time series from the Diffusion model, normalized with (a) sinusoidal normalization and (b) z-score normalization.



(a) GAN Synthesized FEM Sinusoidal Normalized Velocity.

(b) GAN Synthesized FEM Z-Scored Normalized Velocity.

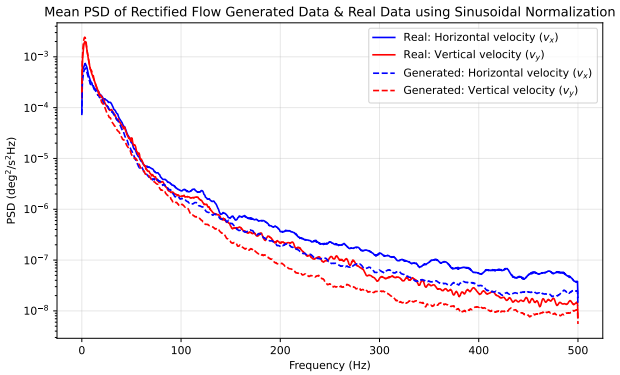
Figure 5.4: Synthesized FEM velocity time series from the GAN model, normalized with (a) sinusoidal normalization and (b) z-score normalization.

5.4 Spectral Analysis

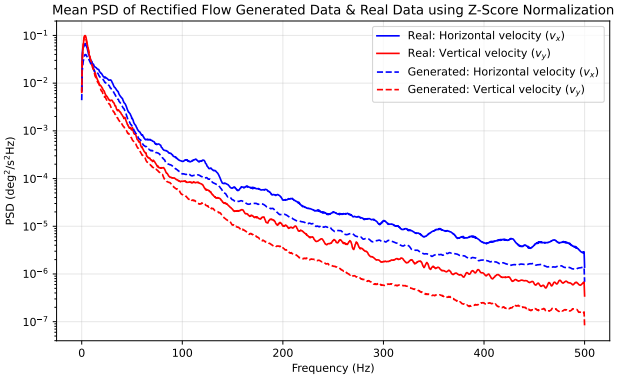
Power Spectral Density (PSD) was computed for real and generated sequences to evaluate frequency domain characteristics. The PSD of real fixation data exhibits a characteristic decay and the Rectified Flow generated sequence, in Figure 5.5, follows that trend closely under both normalizations. In Figure 5.6, the diffusion generated sequences, the Sinusoidal Normalization maintains a comparable power across both low and high frequency bands whilst the Z-score normalization shows overall reduced power suggesting loss of fine scale dynamics. Similarly, the GANs generated sequences also show reduced power across low and high frequency bands failing to reproduce the real data's spectral profile, see Figure 5.7.

5.5 Reliability Analysis

The generated sequences from Rectified Flow, the diffusion and GANs were reproduced across two training session to ensure the evaluation results are reliable. By comparison of the metrics regarding general properties, JSD values and PSD analysis show consistent trends. Models that preserve higher velocity variance also follow the real data spectral profile closely. In addition, the reliability is supported by the use of shared preprocessing pipeline and fixed sequence length. Furthermore, the Rectified flow and diffusion model share the same backbone architecture which ensures that observed differences is not due to architectural size disparities. The use of inter-subject JSD value as a baseline provides an internal consistency check to set reference for model performance relative to variability within the real data.

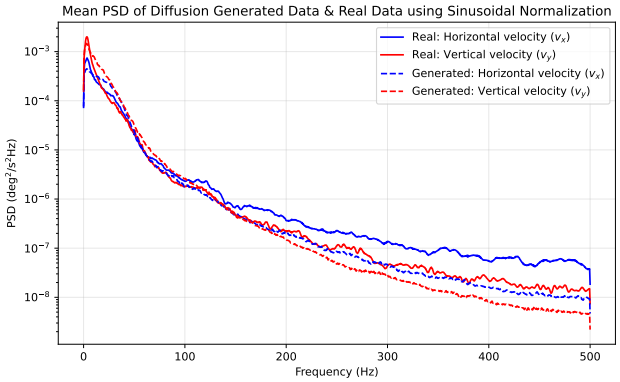


(a) Average PSD of Rectified Flow Synthesized FEM versus Real Data using Sinusoidal Normalized Velocity.

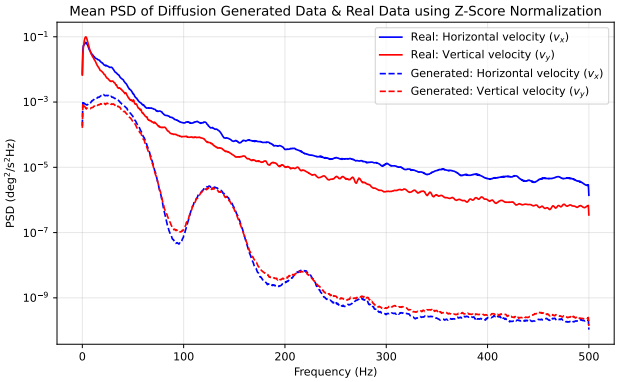


(b) Average PSD of Rectified Flow Synthesized FEM versus Real Data using Sinusoidal Normalized Velocity.

Figure 5.5: Average PSD of synthesized FEM velocity from the Rectified Flow model, normalized with (a) sinusoidal normalization and (b) z-score normalization.

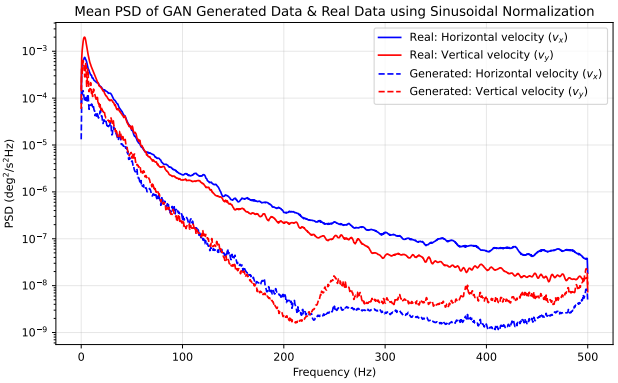


(a) Average PSD of Diffusion Synthesized FEM versus Real Data using Sinusoidal Normalized Velocity.

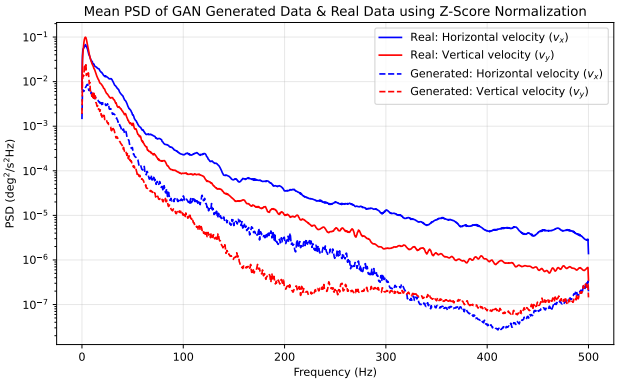


(b) Average PSD of Diffusion Synthesized FEM versus Real Data using Sinusoidal Normalized Velocity.

Figure 5.6: Average PSD of synthesized FEM velocity from the Diffusion model, normalized with (a) sinusoidal normalization and (b) z-score normalization.



(a) Average PSD of GAN Synthesized FEM versus Real Data using Sinusoidal Normalized Velocity.



(b) Average PSD of GAN Synthesized FEM versus Real Data using Sinusoidal Normalized Velocity.

Figure 5.7: Average PSD of synthesized FEM velocity from the GAN model, normalized with (a) sinusoidal normalization and (b) z-score normalization.

5.6 Validity Analysis

The validity of the results are assess through microsaccade characteristics metrics. The microsaccade were detected using Engbert & Kliegl algorithm and following metrics are presented, the average rate of occurring microsaccade and main sequence relationship.

5.6.1 Microsaccade Rate

The average rate of microsaccade across all models as well as the real data is shown in Figure 5.8. The real data exhibited an average microsaccade rate of 2.33 microsaccades per second. In comparison, all three generative approaches produced higher microsaccade rates overall, indicating a general tendency to generate more frequent microsaccade like events than observed in natural FEM.

Among the evaluated models, the Rectified Flow produced a microsaccade rate closest to the real data regardless of the normalization method used. Although still elevated relative to real data, Rectified Flow shows the smallest deviation and therefore best preserved the microsaccade frequency.

The Diffusion model demonstrated the largest mismatch depending on the normalization method. While the Sinusoidal normalization aligns more closely with the actual microsaccade rate compared to the data generated by the Rectified Flow, the Z-Score normalization method substantially degrades

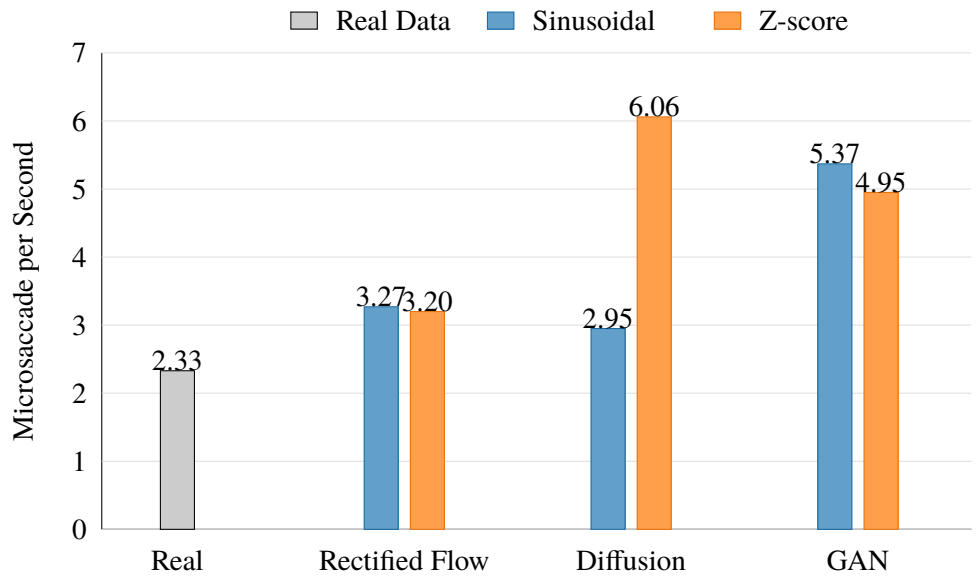


Figure 5.8: The Average rate of Microsaccade per second for Real Data, Rectified Flow, Diffusion, and GAN synthesized FEM.

the performance, representing the strongest over generation of microsaccades among all tested models. This suggests that the samples from Diffusion trained on Z-Score normalized data may contain more high frequency changes that are misinterpreted as microsaccades.

Finally, the GAN model generated consistently high microsaccade rates under both normalization methods, indicating a persistent bias toward producing overly frequent microsaccade events compared to natural FEM.

5.6.2 Main Sequence Relationship

In real FEM, microsaccades follow a characteristic linear trend where larger amplitudes are associated with higher peak velocities. This linear dependence between amplitude and peak velocity is observed in Figure 5.9, with peak velocity increasing approximately proportionally with amplitude and points cluster around the trend with moderate spread.

In Figure 5.10, the Rectified Flow model produced the most physiologically plausible main sequence pattern under both normalization methods, achieving the highest amplitude–peak velocity correlation (Pearson's r = 0.957, $p < 0.001$ for Sinusoidal normalization; r = 0.948, $p < 0.001$ for Z-Score normalization). The synthesized samples exhibited a clear amplitude velocity

dependency with a relatively compact distribution across amplitudes. This indicates that Rectified Flow best preserved the expected coupling between microsaccade size and speed and produced fewer extreme deviations compared to other models.

For the Diffusion model in Figure 5.11, the main sequence relationship was still present but trained on Sinusoidal Normalization, it showed increased spread, particularly at mid-to-large amplitudes. Trained on Z-Score normalization, the diffusion main sequence appeared more compressed in peak velocity, with fewer high velocity events and a denser concentration at mid range velocities, suggesting reduced dynamic range compared to the sinusoidal case. Overall, diffusion model exhibited stronger dependence on preprocessing.

The GAN model produced the weakest match to the expected main sequence structure, see Figure 5.12. While the amplitude velocity increase was visible, the distribution showed substantial dispersion and a flatter slope, which is less consistent with microsaccade dynamics. This effect was especially noticeable as a broad spread at larger amplitudes, indicating that the GAN frequently generated large amplitude events without corresponding increases in peak velocity. This suggests that the GAN struggles to preserve realistic microsaccade dynamics.

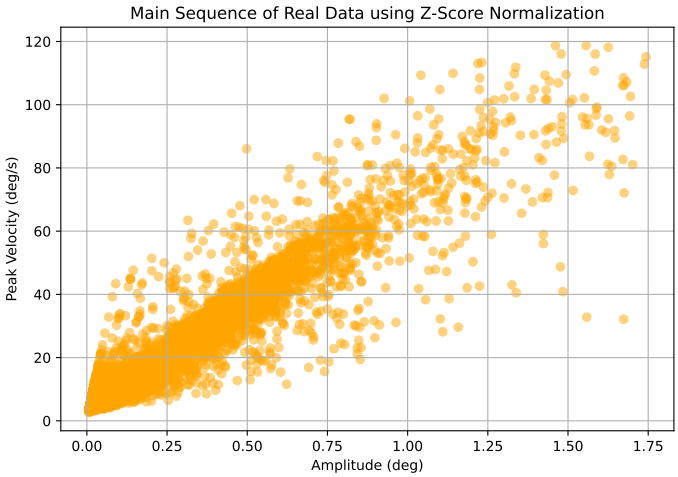
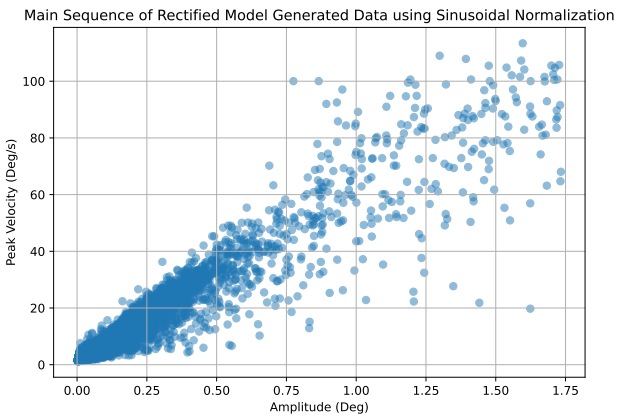
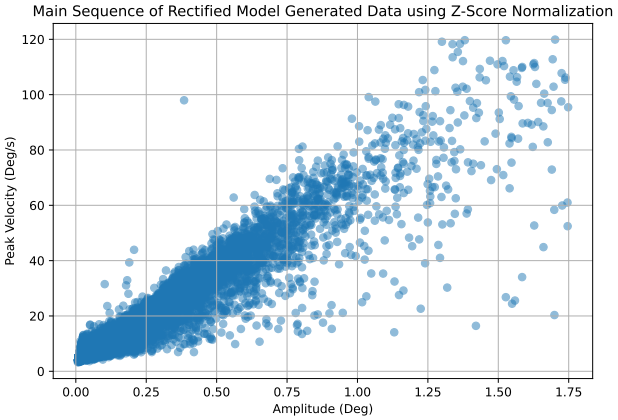


Figure 5.9: Main Sequence of Real FEM.

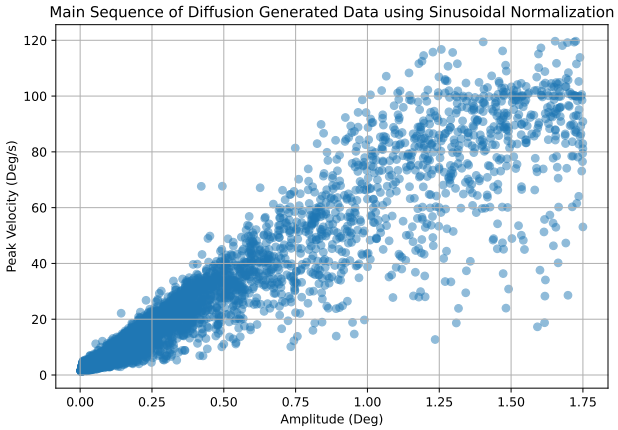


(a) Main Sequence of Rectified Flow Synthesized FEM using Sinusoidal Normalization.

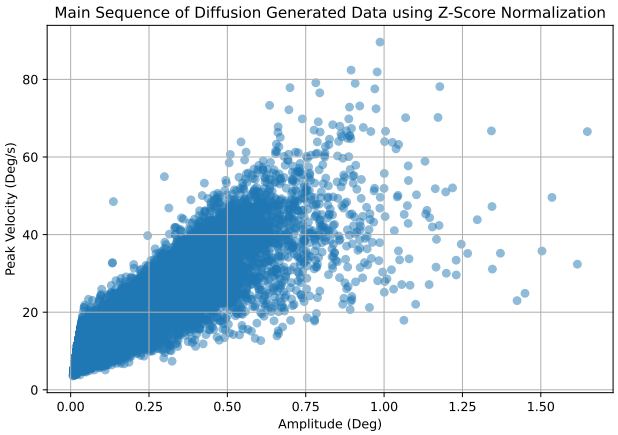


(b) Main Sequence of Rectified Flow Synthesized FEM using Z-Score Normalization.

Figure 5.10: Main Sequence of synthesized FEM velocity from the Rectified Flow model, normalized with (a) sinusoidal normalization and (b) z-score normalization.

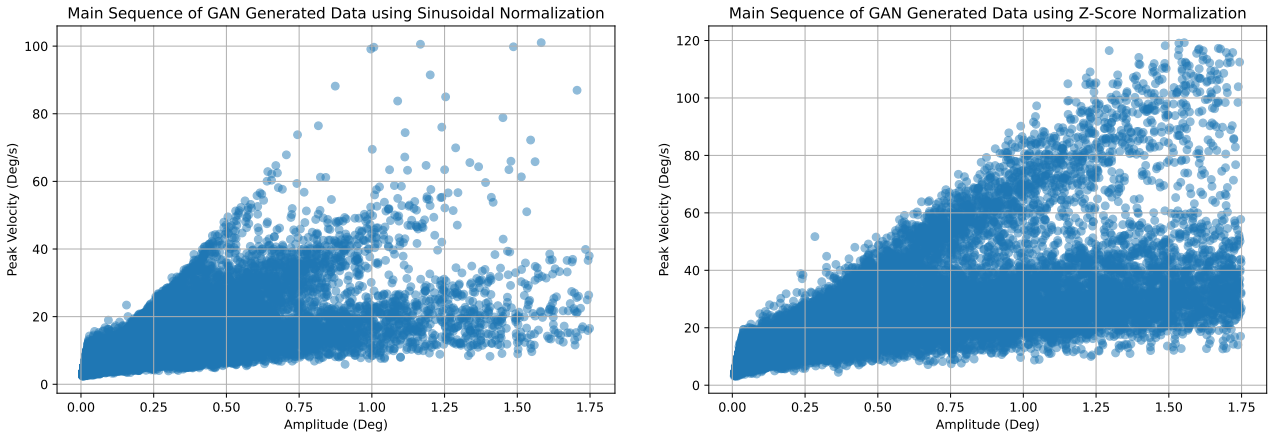


(a) Main Sequence of Diffusion Synthesized FEM using Sinusoidal Normalization.



(b) Main Sequence of Diffusion Synthesized FEM using Z-Score Normalization.

Figure 5.11: Main Sequence of synthesized FEM velocity from the Diffusion model, normalized with (a) sinusoidal normalization and (b) z-score normalization.



(a) Main Sequence of GAN Synthesized FEM using Sinusoidal Normalization.

(b) Main Sequence of GAN Synthesized FEM using Z-Score Normalization.

Figure 5.12: Main Sequence of synthesized FEM velocity from the GAN model, normalized with (a) sinusoidal normalization and (b) z-score normalization.

5.7 Summary

Overall, the Rectified Flow model consistently show improved results across multiple metrics compared to Diffusion and GANs. Although in some instances under Sinusoidal Normalization the Diffusion generated sequences outperforms Rectified Flow model. However, it is clear that the Z-score normalization degrades the performance of the Diffusion model. In comparison, for Rectified Flow model the choice of normalization method demonstrated minimal impact on the generated results. Finally, GANs generated sequence does not consistently fully capture the FEM dynamic resulting in noisy generated sequences.

Chapter 6

Discussion

This thesis investigated the unconditional generation of FEM using three generative models: diffusion, GANs and Rectified Flow. The key contribution of the thesis was the application of Rectified Flow as part of Flow Matching to the domain of eye tracking. The result demonstrate the Rectified Flow outperforms diffusion and GANs in producing synthetic FEM across distributional similarity, spectral and physiological metrics. However, none of the evaluated models fully reproduced all characteristics of real FEM. Although several metrics suggest statistical similarity, physiological realism requires preserving microsaccade behavior. Microsaccades follow constrained occurrence rates and a structured main sequence, reflecting oculomotor control mechanisms. Therefore, discrepancies in microsaccade rate or main sequence structure indicate that a model may match global distributions while failing to reproduce biologically valid FEM. This is particularly relevant for clinical applications, where unrealistic microsaccade dynamics may bias subsequent analyses. Following this, in this chapter, the implication of results are discussed.

6.1 Interpretation of Microsaccade Characteristics

Microsaccade characteristics provide a physiologically grounded evaluation of synthetic FEM, since they reflect both the structure and the underlying oculomotor control constraints. While other metrics (e.g., general signal properties, PSD and JSD) are useful for assessing overall signal realism,

they do not guarantee that the generated data preserve biologically plausible fixation. In this context, microsaccade rate and main sequence represent more informative metric because they directly measure whether the generated sequences reproduce known regularities of natural FEM.

A key observation from the results were that all generated samples exhibited higher average rate of microsaccades. Natural FEM typically contains microsaccades rate of approximately 2 events per second, and deviations from this indicate that the model is likely producing either excessive transient events or insufficient microsaccadic corrections. Overgeneration is problematic because it introduces an unrealistic dominance of microsaccades relative to drift and tremor components. This suggests that the model has not learned the correct behavior and may instead rely on high frequency fluctuations that are interpreted as microsaccades. As a result, even if the synthesized traces appear visually similar or match power spectra, their physiological validity remains limited.

The main sequence relationship further reinforces the importance of microsaccade evaluation. In real FEM, microsaccades exhibit a strong positive and approximately linear coupling between amplitude and peak velocity. Preserving this relationship is essential because it indicates that the model does not merely produce abrupt jumps, but rather generates microsaccades with internally consistent dynamics. A weakened or highly dispersed main sequence pattern implies that the model does not consistently reproduce the scaling between microsaccade size and speed, and instead generates events with unrealistic velocity profiles. Such deviations reduce the reliability of the synthesized data for downstream analyses that depend on microsaccade dynamics.

This is particularly relevant for Parkinson's disease applications, where fixation abnormalities are often reflected in microsaccade behavior. Synthetic signals that distort microsaccade frequency or main sequence risk introducing artifacts that may bias downstream clinical interpretation.

6.2 Distributional Similarity as Metric

An important observation from the results is that JSD value do not necessarily imply equivalent physiological realism. For example, the diffusion model achieves similar JSD value despite clear difference in temporal variability, spectral profile and microsaccade characteristics. This highlights a limitation

of histogram based distributional as a metric, since it captures marginal similarity but is insensitive to temporal structure and microsaccade events. Thus motivating the choice of evaluating the microsaccade characteristics to validate the synthetic FEM.

6.3 Rectified Flow as a Novel Approach for FEM Synthesis

A key contribution of this project is the application of Rectified Flow model to FEM synthesis. Rectified Flow consistently achieves lowest JSD value across both normalization methods and produced stable spectral profile matching the real data. Under Z-Score normalization, the extracted microsaccade matches closest to the real data, suggesting Z-Score normalization appears more suitable under the tested conditions for Rectified Flow on FEM synthesis.

Unlike Diffusion models, Rectified Flow does not rely on stochastic denoising across discrete timestep but instead learns a deterministic velocity field that transport samples from noise distribution to the data distribution. Based on the results, this formulation seems to be well suited for modeling the global structure of fixation velocity distributions. Furthermore, Rectified Flow sample generation is done at significantly lower steps than compared to Diffusion model.

An additional factor contributing to the strong performance of Rectified Flow model is the stability of its training objective. Although both diffusion and Rectified Flow share the same DiffWave backbone, their optimization behavior differ due to the training formulation which affect the regularization strategy. In this work, the Diffusion model saw no improvement using weight decay or not, therefore the setup that is more common in Diffusion based denoising objective was chosen. In contrast, Rectified Flow using weight decay and cosine annealing led to more stable training. This benefit can be due to how Rectified Flow directly regresses a deterministic velocity mapping between noise and data and without regularization this mapping can become sensitive to local variations in training data.

Overall the results indicate that Rectified Flow provides the most physiologically plausible microsaccade behavior, as it produces rates closest to the natural range and maintains a clear main sequence trend with comparatively low dispersion. Importantly, its microsaccade characteristics remain stable across normalization strategies. Nevertheless, minor deviations

in microsaccade characteristic indicate that the Rectified Flow model does not fully reproduce all physiological constraints, highlighting the remaining gap between statistical similarity and complete oculomotor realism.

6.4 Diffusion Model

Diffusion models show strong overall performance, but their behavior is highly dependent on the choice of preprocessing. Sinusoidal Normalization consistently produces more realistic fixation dynamic than Z-Score Normalization. This observation aligns also with findings reported in DiffEyeSyn [7] where the authors noted that Z-Score Normalization tends to oversmooth Diffusion generated eye movement signals. This sensitivity suggests that Diffusion model may be more dependent on the signal scaling used during training, which can alter the amount of transient events relative to background drift. As a result, preprocessing can substantially influence whether the model preserves realistic fixational dynamics or produces exaggerated microsaccade like fluctuations. This was observed in microsaccade rate becoming substantially inflated and the main sequence structure becoming less consistent with real FEM.

While Rectified Flow demonstrates strong overall performance and robustness across normalizations, the results also show that Diffusion models trained under Sinusoidal Normalization achieve strong performance and in some metrics similar to Rectified Flow.

Overall, the results suggest Diffusion is dependent on preprocessing choice and paired with Sinusoidal Normalization it can achieve comparable results to Rectified Flow in metrics sensitive to temporal dynamics.

6.5 GAN Model

GANs generated sequence consistency underperforms diffusion and Rectified Flow model across most evaluation metrics. Despite having a significantly larger parameter count, exhibits significant amount of dispersion in microsaccade characteristics. The wide spread observed in main sequence plot indicated weak relation between amplitude and peak velocity, suggesting that GANs struggle to capture the structured relationship inherent in FEM dynamic. This behavior is also consistent with lower spectral power profile and higher JSD value. This suggests that adversarial training does not

consistently captures local variation, leading to unstable microsaccade timing and inconsistent dynamic.

Chapter 7

Conclusions and Future work

In summary, this thesis evaluated the use of state of the art generative models for unconditional FEM synthesis, focusing on novel application of Rectified Flow in comparison to baselines methods of diffusion and GANs. The evaluation combined, general temporal properties, spectral and physiological metrics including microsaccade rate and main sequence to asses both distributional similarity and FEM realism validity.

Based on the evaluation of the results and in addressing this thesis's research question, it can be concluded that Rectified Flow outperforms the baseline models overall. Therefore the goals of this project are met. Rectified Flow provides the strongest physiological validity, Diffusion is comparable but preprocessing sensitive, and GANs are least suitable for realistic FEM synthesis.

Another key insight gained in this work, is that the distributional metrics such as JSD are alone insufficient to validate the realism of generated FEM as they do not capture temporal and physiological structure. This motivated the evaluation of microsaccade characteristics. Overall, the results highlight that microsaccade characteristics should be prioritized when evaluating generative FEM models. Achieving a realistic microsaccade rate within the expected physiological range and reproducing a strong main sequence relationship provides stronger evidence of biological plausibility than global similarity metrics alone.

7.1 Limitations

In this work, all experiment were conducted using only Gazebase Fixation Task and as a results the generalization of the finding to other dataset cannot be directly established.

A key limitation of this project is that the microsaccade characteristic evaluation indicates that none of the generative models fully reproduce physiologically realistic microsaccade dynamics. In particular, the generated sequences exhibit inflated microsaccade rates and, in some cases, a weakened or overly dispersed main-sequence relationship, suggesting that the models may capture global signal structure while failing to consistently preserve finer microsaccade events.

Finally, this thesis only provided a baseline of models with focus on similar parameter size with the exception of GANs, and no extensive hyperparameter search or architecture specific optimization were conducted. Therefore the performances of the models should be interpreted as representative but not necessarily as fully optimal.

7.2 Future work

As main extension from this work, model selection and future improvements should focus on reducing microsaccade overgeneration and maintaining a stronger amplitude velocity coupling, as these properties are essential for producing synthetic FEM that is not only visually plausible but also consistent with established oculomotor behavior.

It would also be of interest to synthesize longer samples than 5 seconds. A possible method could be a similar approach done in SP-EyeGAN [6], where multiple short sequences are generated and stitched together. On the same matter, future work could also explore conditional generation with emphasis on incorporation physiological constraint directly into training objective.

On a final note, the project's original idea was to address scarce clinical data for training of Early Parkinson Disease Classifier, it would be valuable to evaluate whether the generated FEM do in fact translate into measurable performance gain for the downstream task.

References

[1] I. Ahmad and F. Alqurashi, "Early cancer detection using deep learning and medical imaging: A survey," *Critical Reviews in Oncology/Hematology*, vol. 204, p. 104528, 2024. doi: https://doi.org/10.1016/j.critrevonc.2024.104528. [Online]. Available: https://www.sciencedirect.com/science/article/pii/S1040842824002713

[2] M. Rolfs, "Microsaccades: Small steps on a long way," *Vision Research*, vol. 49, no. 20, pp. 2415–2441, 2009. doi: https://doi.org/10.1016/j.visres.2009.08.010. [Online]. Available: https://www.sciencedirect.com/science/article/pii/S0042698909003691

[3] J. Ho, A. Jain, and P. Abbeel, "Denoising diffusion probabilistic models," 2020. [Online]. Available: https://arxiv.org/abs/2006.11239

[4] Paper Digest, "Advances in flow matching: Insights from icml 2025 papers," https://www.paperdigest.org/report/?id=advances-in-flow-matching-insights-from-icml-2025-papers, 2025, last updated: 2025-06-25. [Online]. Available: https://www.paperdigest.org/report/?id=advances-in-flow-matching-insights-from-icml-2025-papers

[5] X. Zhang, Y. Pu, Y. Kawamura, A. Loza, Y. Bengio, D. L. Shung, and A. Tong, "Trajectory flow matching with applications to clinical time series modeling," 2025. [Online]. Available: https://arxiv.org/abs/2410.21154

[6] P. Prasse, D. R. Reich, S. Makowski, S. Ahn, T. Scheffer, and L. A. Jäger, "Sp-eyegan: Generating synthetic eye movement data with generative adversarial networks," in *Proceedings of the 2023 Symposium on Eye Tracking Research and Applications*, ser. ETRA '23. New York, NY, USA: Association for Computing Machinery, 2023. doi:

10.1145/3588015.3588410. ISBN 9798400701504. [Online]. Available: https://doi.org/10.1145/3588015.3588410

[7] C. Jiao, G. Zhang, Y. Cho, Z. Hu, and A. Bulling, "Diffeyesyn: Diffusion-based user-specific eye movement synthesis," 2025. [Online]. Available: https://arxiv.org/abs/2409.01240

[8] S. Martinez-Conde, S. L. Macknik, and D. H. Hubel, "The role of fixational eye movements in visual perception," *Nature Reviews Neuroscience*, vol. 5, no. 3, pp. 229–240, 2004. doi: 10.1038/nrn1348

[9] R. Engbert and R. Kliegl, "Microsaccades uncover the orientation of covert attention," *Vision Research*, vol. 43, no. 9, pp. 1035–1045, 2003. doi: https://doi.org/10.1016/S0042-6989(03)00084-1. [Online]. Available: https://www.sciencedirect.com/science/article/pii/S0042698903000841

[10] J. Lin, "Divergence measures based on the shannon entropy," *IEEE Transactions on Information Theory*, vol. 37, no. 1, pp. 145–151, 1991. doi: 10.1109/18.61115

[11] J. Song, C. Meng, and S. Ermon, "Denoising diffusion implicit models," 2022. [Online]. Available: https://arxiv.org/abs/2010.02502

[12] I. J. Goodfellow, J. Pouget-Abadie, M. Mirza, B. Xu, D. Warde-Farley, S. Ozair, A. Courville, and Y. Bengio, "Generative adversarial networks," 2014. [Online]. Available: https://arxiv.org/abs/1406.2661

[13] A. Radford, L. Metz, and S. Chintala, "Unsupervised representation learning with deep convolutional generative adversarial networks," 2016. [Online]. Available: https://arxiv.org/abs/1511.06434

[14] Y. Lipman, R. T. Q. Chen, H. Ben-Hamu, M. Nickel, and M. Le, "Flow matching for generative modeling," 2023. [Online]. Available: https://arxiv.org/abs/2210.02747

[15] X. Liu, C. Gong, and Q. Liu, "Flow straight and fast: Learning to generate and transfer data with rectified flow," 2022. [Online]. Available: https://arxiv.org/abs/2209.03003

[16] Z. Kong, W. Ping, J. Huang, K. Zhao, and B. Catanzaro, "Diffwave: A versatile diffusion model for audio synthesis," 2021. [Online]. Available: https://arxiv.org/abs/2009.09761

[17] L. Bolliger, D. Reich, P. Haller, D. Jakobi, P. Prasse, and L. Jäger, "Scandl: A diffusion model for generating synthetic scanpaths on texts," 01 2023. doi: 10.18653/v1/2023.emnlp-main.960 pp. 15 513–15 538.

[18] L. S. Bolliger, D. R. Reich, and L. A. Jäger, "Scandl 2.0: A generative model of eye movements in reading synthesizing scanpaths and fixation durations," *Proc. ACM Hum.-Comput. Interact.*, vol. 9, no. 3, May 2025. doi: 10.1145/3725830. [Online]. Available: https://doi.org/10.1145/3725830

[19] Y. Hu, X. Wang, Z. Ding, L. Wu, H. Zhang, S. Z. Li, S. Wang, J. Zhang, Z. Li, and T. Chen, "Flowts: Time series generation via rectified flow," 2025. [Online]. Available: https://arxiv.org/abs/2411.07506

[20] H. Griffith, D. Lohr, E. Abdulin, and et al., "GazeBase, a large-scale, multi-stimulus, longitudinal eye movement dataset," *Scientific Data*, vol. 8, p. 184, 2021. doi: 10.1038/s41597-021-00959-y. [Online]. Available: https://doi.org/10.1038/s41597-021-00959-y

[21] S. Makowski, L. A. Jäger, and T. Scheffer, "Judo1000 eye tracking data set," OSF, 2020. [Online]. Available: https://doi.org/10.17605/OSF.IO/5ZPVK

[22] D. G. Krakowczyk, D. R. Reich, J. Chwastek, D. N. Jakobi, P. Prasse, A. Süss, O. Turuta, P. Kasprowski, and L. A. Jäger, "pymovements: A python package for processing eye movement data," in *2023 Symposium on Eye Tracking Research and Applications*, ser. ETRA '23. New York, NY, USA: Association for Computing Machinery, 2023. doi: 10.1145/3588015.3590134. ISBN 979-8-4007-0150-4/23/05. [Online]. Available: https://doi.org/10.1145/3588015.3590134

[23] "Naiss: National ai and supercomputing infrastructure," https://www.naiss.se/, accessed: 2025-12-19.

[24] I. Gulrajani, F. Ahmed, M. Arjovsky, V. Dumoulin, and A. Courville, "Improved training of wasserstein gans," 2017. [Online]. Available: https://arxiv.org/abs/1704.00028

[25] S. Lee, Z. Lin, and G. Fanti, "Improving the training of rectified flows," 2024. [Online]. Available: https://arxiv.org/abs/2405.20320

TRITA – XXX-EX 2025:0000
Stockholm, Sweden 2026

www.kth.se

€€€€ For DIVA €€€€

{
”Author1”: { ”Last name”: ”Safdari”,
”First name”: ”Rahnama R.”,
”Local User Id”: ”u1ucq7d0”,
”E-mail”: ”rahnamas@kth.se”,
”organisation”: {”L1”: ”School of Electrical Engineering and Computer Science”,
}
},
”Cycle”: ”2”,
”Course code”: ”DA233X”,
”Credits”: ”30”,
”Degree1”: {”Educational program”: ”Degree Programme in Electrical Engineering”
,”programcode”: ”CELTE”
,”Degree”: ”Degree of Master of Science in Engineering”
,”subjectArea”: ”Machine Learning”
},
”Title”: {
”Main title”: ”Evaluating Rectified Flow Against Diffusion and GANs Models for Synthetic Generation of Fixational Eye Movements”,
”Subtitle”: ”Benchmarking Rectified Flow, diffusion, and GANs for fixation eye movement generation”,
”Language”: ”eng” },
”Alternative title”: {
”Main title”: ”Utvärdering av Rectified Flow jämfört med diffusion- och GAN-modeller för syntetisk generering av ögonfixationer”,
”Subtitle”: ”Benchmarking av Rectified Flow, diffusion-modeller och GANs för generering av ögonfixationer”,
”Language”: ”swe”
},
”Supervisor1”: { ”Last name”: ”Saketos”,
”First name”: ”Vasileios”,
”Local User Id”: ”u103t48q”,
”E-mail”: ”(saketos@kth.se”,
”organisation”: {”L1”: ”School of Electrical Engineering and Computer Science”,
”L2”: ”Division of Information Science and Engineering” }
},
”Supervisor2”: { ”Last name”: ”Wang”,
”First name”: ”Yiting”,
”E-mail”: ”yiting.wang@ki.se”,
”Other organisation”: ”Karolinska Institute, Department of Clinical Neuroscience, Division of Eye and Vision”
},
”Supervisor3”: { ”Last name”: ”Nilsson”,
”First name”: ”Mattias”,
”E-mail”: ”mattias.nilsson@ki.se”,
”Other organisation”: ”Karolinska Institute, Department of Clinical Neuroscience, Division of Eye and Vision”
},
”Examiner1”: { ”Last name”: ”Payberah”,
”First name”: ”Amir H.”,
”Local User Id”: ”u1a73o9d”,
”E-mail”: ”payberah@kth.se”,
”organisation”: {”L1”: ”School of Electrical Engineering and Computer Science”,
”L2”: ”Division of Software and Computer Systems” }
},
"National Subject Categories": "ddddd, ddddd",
"SDGs": "XXX, XXX",
”Other information”: {”Year”: ”2026”, ”Number of pages”: ”xv,59”},
”Copyrightleft”: ”copyright”,
”Series”: { ”Title of series”: ”TRITA – XXX-EX” , ”No. in series”: ”2025:0000” },
”Opponents”: { ”Name”: ”XXX”},
”Presentation”: { ”Date”: ”2022-03-15 13:00”

,”Language”:”XXX”

,”Room”: ”via Zoom https://kth-se.zoom.us/j/ddddddddddd”

,”Address”: ”Isafjordsgatan 22 (Kistagången 16)”

,”City”: ”Stockholm” },

”Number of lang instances”: ”2”,
”Abstract[eng]”: €€€€
Fixational eye movements are involuntary eye motions occurring during visual fixation and consist primarily of drift, tremor, and microsaccades. These movements have been studied as potential markers for neurological disorders such as ’Parkinsons disease. Despite this relevance, large scale datasets for fixational eye movement are scarce, particularly in clinical settings. This thesis addresses the problem of synthesizing realistic fixational eye movements using deep generative models. Generating physiologically plausible fixation signals is challenging because fixational eye movement exhibits both stochastic variability and structured microsaccade characteristic such as event rates and a main sequence relationship. Prior approaches often prioritize global signal similarity and downstream task, but may be limited to reproduce real microsaccade characteristics, reducing their reliability for downstream analysis and data augmentation. To investigate this problem, three generative methods

were implemented, Generative Adversarial Networks, diffusion based models, and novel application of Rectified Flow. All models were trained on preprocessed positional recordings using two normalization methods, Sinusoidal normalization and Z-Score normalization, and generated samples were reverse normalized for microsaccade evaluation. Model performance was assessed through a combination of signal level and physiological metrics, including distributional similarity measures, power spectral density analysis, and microsaccade based evaluation. The results show that Rectified Flow generation generation is closest to the real data and provides the most consistent performance across evaluation metrics and demonstrates robustness to the choice of normalization method. Diffusion based generation achieves competitive performance under Sinusoidal normalization but degrades under Z-Score normalization, suggesting sensitivity to preprocessing that impacts temporal dynamics. In contrast, Generative Adversarial Networks produce noisier sequences and exhibit weaker physiological microsaccade characteristics. Overall, the findings highlight that microsaccade characteristics provide an important validity criterion beyond global similarity measures and support Rectified Flow as a promising approach for generating physiologically plausible fixational eye movement. Although Rectified Flow achieved the strongest overall performance, the generated sequences still did not fully match all microsaccade characteristics observed in real data, highlighting remaining limitations in physiological realism. Future work should therefore focus on improving microsaccade realism and validating model generalization across datasets.

"Abstract[swe]": €€€€
Fixeringsögonrörelser är ofrivilliga ögonrörelser som uppstår under visuell fixering och består huvudsakligen av drift, tremor och mikrosaccad. Dessa rörelser har studerats som potentiella markörer för neurologiska sjukdomar såsom Parkinsons sjukdom. Trots denna relevans är storskaliga datamängder för fixeringsögonrörelser begränsad i storlek, särskilt i kliniska miljöer. Denna arbete behandlar problemet med att syntetisera realistiska fixeringsögonrörelser med hjälp av djupa generativa modeller. Att generera fysiologiskt rimliga fixationssignaler är utmanande eftersom fixeringsögonrörelser uppvisar både stokastisk variabilitet och strukturerad mikrosaccad karakteristik såsom händelsefrekvenser och ett huvudsekvens förhållande. Tidigare metoder prioriterar ofta global signallikhet och nedströmsuppgift, men kan vara begränsade till att reproducera verkliga mikrosaccad karakteristik, vilket minskar deras tillförlitlighet för nedströmsanalys och dataförstärkning. För att undersöka detta problem implementerades tre generativa metoder, Generativa Adversarial Nätverk, diffusionsbaserade modeller och en ny tillämpning av Rectified Flow. Alla modeller tränades på förbearbetade positionsregistreringar med hjälp av två normaliseringsmetoder, sinusformad normalisering och Z-Score-normalisering, och genererade data omvänt normaliserades för mikrosaccad utvärdering. Modellprestanda utvärderades genom en kombination av signalnivå och fysiologiska mätvärden, inklusive distributionslikhetsmått, effektspektraldensitetsanalys och mikrosaccad baserad utvärdering. Resultaten visar att Rectified Flow-generering är närmast verkliga data och ger den mest konsekventa prestandan över utvärderingsmått samt demonstrerar robusthet i valet av normaliseringsmetod. Diffusionsbaserad generering uppnår nära prestanda under sinusformad normalisering men försämras under Z-Score-normalisering, vilket tyder på känslighet för förbehandling som påverkar temporal dynamik. Däremot producerar Generativa Adversarial Nätverk mer brusiga sekvenser och uppvisar svagare fysiologiska mikrosaccad egenskaper. Sammantaget belyser resultaten att mikrosaccad egenskaper ger ett viktigt validitetskriterium utöver globala likhetsmått och stöder Rectified Flow som en lovande metod för att generera fysiologiskt verkliga fixeringsögonrörelser. Även om Rectified Flow uppnådde den starkaste totala prestandan, matchade de genererade sekvenserna fortfarande inte helt alla mikrosaccad egenskaper som observerats i verkliga data, vilket belyser återstående begränsningar i fysiologisk realism. Framtida arbete bör därför fokusera på att förbättra mikrosaccad realismen och validera modellgeneralisering över datamängder.

€€€€,
"Keywords[eng]": €€€€
Deep Generative Models, Rectified Flow, Flow Matching, Diffusion Models, Generative Adversarial Networks, Fixational Eye Movements, Synthetic Data Generation, Continuous-Time Generative Models €€€€,
€€€€,
"Keywords[swe]": €€€€
Djupa generativa modeller, Rektifierat flöde, Flödesmatchning, Diffusionsmodeller, Generativa adversariella nätverk, Fixationella ögonrörelser, Syntetisk datagenerering, Kontinuerliga tidsbaserade generativa modeller €€€€,
}

acronyms.tex

```
%%% Local Variables:
%%% mode: latex
%%% TeX-master: t
%%% End:
% The following command is used with glossaries-extra
\setabbreviationstyle[acronym]{short}
% The form of the entries in this file is \newacronym{label}{acronym}{phrase}
%                                     or \newacronym[options]{label}{acronym}{phrase}
% see "User Manual for glossaries.sty" for the  details about the options, one example is shown below
% note the specification of the long form plural in the line below
% The following example also uses options
% note the use of a non-breaking dash in long text for the following acronym
% example of putting in a trademark on first expansion

\newacronym{AI}{AI}{Artificial Intelligence}
\newacronym{DCGANs}{DCGANs}{Deep Convolutional Generative Adversarial Networks}
\newacronym{DDIM}{DDIM}{Denoising Diffusion Implicit Models}
\newacronym{DDPM}{DDPM}{Denoising Diffusion Probabilistic Models}
\newacronym{FEM}{FEM}{Fixational Eye Movement}
\newacronym{GANs}{GANs}{Generative Adversarial Networks}
\newacronym{JSD}{JSD}{Jensen Shannon Divergence}
\newacronym{ReLU}{ReLU}{Rectified Linear Unit}
\newacronym{ODE}{ODE}{Ordinary Differential Equation}
\newacronym{SDE}{SDE}{Stochastic Differential Equation}
\newacronym{MLP}{MLP}{Multilayer Perceptron}
\newacronym{SiLU}{SiLU}{Sigmoid Linear Unit}
\newacronym{PSD}{PSD}{Power Spectral Density}
```